2004-03

Predicting catastrophic BGP routing instabilities

Nguyen, Lien K.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

PREDICTING CATASTROPHIC BGP ROUTING INSTABILITIES

by

Lien K. Nguyen

March 2004

Thesis Advisor: Geoffrey Xie
Second Reader: J.D. Fulp

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 2004	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE: Predicting Catastrophic BGP Routing Instabilities	5. FUNDING NUMBERS
6. AUTHOR(S) Lien K. Nguyen	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

Inter-domain routing connects individual pieces of Internet topology, creating an integral, global data delivery infrastructure. Currently, this critical function is performed by the Border Gateway Protocol (BGP) version 4 [RFC1771]. Like all routing protocols, BGP is vulnerable to instabilities that reduce its effectiveness. Among the causes of these instabilities are those which are maliciously induced. Although there are other causes, e.g., natural events and network anomalies, this thesis will focus exclusively on maliciously induced instabilities.

Most current models that attempt to predict a BGP routing instability confine their focus to either macro- or micro-level metrics, but not to both. The inherent limitations of each of these forms of metric gives rise to an excessive rate of spurious alerts, both false positives and false negatives. It is the original intent of this thesis to develop an improved BGP instability prediction model by statistically combining BGP instability metrics with user level performance metrics. The motivation for such a model is twofold. 1) To provide sufficient prior warning of impending failure to facilitate proactive protection measures. 2) To improve warning reliability beyond existing models, by demonstrably reducing both false positives and false negatives. However, our analysis of actual network trace data shows that a widely used BGP instability metric, the total number of update messages received in a time period, is not a good indicator of future user level performance.

14. SUBJECT TERMS: Catastrophic BGP Routing Instabilities; Macro-Level and Micro-Level Metrics Correlation; Worm Attack Studies;	15. NUMBER OF PAGES 171
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

PREDICTING CATASTROPHIC BGP ROUTING INSTABILITIES

Lien K. Nguyen
Civilian, Department of Defense
B.S., California State University of Long Beach, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
March 2004

Author: Lien Kim Nguyen

Approved by: Geoffrey Xie
Thesis Advisor

J.D. Fulp
Second Reader

Peter Denning
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Inter-domain routing connects individual pieces of Internet topology, creating an integral, global data delivery infrastructure. Currently, this critical function is performed by the Border Gateway Protocol (BGP) version 4 [RFC1771]. Like all routing protocols, BGP is vulnerable to instabilities that reduce its effectiveness. Among the causes of these instabilities are those which are maliciously induced. Although there are other causes, e.g., natural events and network anomalies, this thesis will focus exclusively on maliciously induced instabilities.

Most current models that attempt to predict a BGP routing instability confine their focus to either macro- or micro-level metrics, but not to both. The inherent limitations of each of these forms of metric gives rise to an excessive rate of spurious alerts, both false positives and false negatives. It is the original intent of this thesis to develop an improved BGP instability prediction model by statistically combining BGP instability metrics with user level performance metrics. The motivation for such a model is twofold. 1) To provide sufficient prior warning of impending failure to facilitate proactive protection measures. 2) To improve warning reliability beyond existing models, by demonstrably reducing both false positives and false negatives. However, our analysis of actual network trace data shows that a widely used BGP instability metric, the total number of update messages received in a time period, is not a good indicator of future user level performance.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I want to take this opportunity to express my profound gratitude to my thesis advisor, Dr. Geoffrey Xie, for his constant devotion to helping me solve problems and giving me ideas on how to perform independent research.

I would like to thank: my editor/co-advisor, Victoria J. Galante, for her time and fast response; J.D Fulp for helping me get a good start on this thesis; Prof. Bob Broadston, John Clark and Nathan Beltz for their time and ideas on technical areas;

I would like to thank Dr. Irvine for working hard to make this SFS program a rigorous and valuable program.

I would also like to thank the following staff members: Prof. Daniel Warren, Tanya Raven, Deborah Shifflett, Jean Brennan, Dr. Thomas Otani for directing, guiding me and other SFS students, and making life as a researcher at NPS much easier.

Finally, I would like to thank the NSF and the NSA for giving me and other SFS students such a wonderful opportunity to do research in the field of information assurance. Thanks for giving me a great chance to serve our country's needs. The bottom line is I am very happy and grateful to receive a Master's degree in Computer Science.

This material is based on work supported by the National Science Foundation under Grant DUE-0210762.

This material is based on work supported by the National Security Agency.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. RESEARCH OBJECTIVES

The objective of this thesis, and of the research upon which it is based, is to determine whether the combination of macro-level (user-observable) and micro-level (network-event) metrics yields a two-fold benefit: 1) to increase the interval between warning and the catastrophic event, and 2) to capture more catastrophic events with fewer false-alerts than are presently detected by conventional macro-only or micro-only metric-based methods. This thesis is concerned only with one kind of catastrophic event: those perpetrated with malicious intent.

B. SCOPE AND ASSUMPTIONS

This thesis will concentrate on three main issues related to BGP instability prediction: 1) Collection/marshalling existing data. 2) Analysis and correlation of existing data. 3) Testing of one or more developed correlation models to identify the model that is both: a) the most reliable (i.e., yields the fewest false-positives and false-negatives), and b) generates the greatest advance warning. This thesis is not responsible for generating new BGP instability data; i.e., no prototype/test network will be configured to generate or collect new data. Lacking any catastrophe-causing instability during the time of thesis, we will use the data collected during the Blaster Worm and the East Coast Blackout periods that widely infected the Internet in August 2003.

C. METHODOLOGY

Research progressed in four main phases: 1) Review of current methods of measuring BGP Routing Instabilities. 2) Identification of the most effective micro-level and macro-level measures of BGP Routing Instabilities. 3) Collection of data. 4) Development of a statistical model combining the collected data to accurately predict catastrophic BGP routing instabilities.

D. RESEARCH QUESTIONS

The primary goal of this thesis is to answer the following questions:

- Primary research question: What is the best way to accurately predict potentially catastrophic Border Gateway Protocol (BGP) routing instabilities?
- Subsidiary research question #1: What is the basic operational description of the BGP's function as an inter-autonomous system (AS) routing protocol?
- Subsidiary research question #2: Identify the existing causes and definitions of malicious BGP routing instabilities.
- Subsidiary research question #3: What are the definitions of, and the distinction between macro- and micro-level measures of BGP routing instabilities?
- Subsidiary research question #4: Identify the existing macro-level measures of BGP routing instabilities.
- Subsidiary research question #5: Identify the existing micro-level measures of BGP routing instabilities.
- Subsidiary research question #6: How is data for macro-level measures of BGP routing instabilities collected?
- Subsidiary research question #7: How is data for micro-level measures of BGP routing instabilities collected?
- Subsidiary research question #8: How can the data derived from subsidiary research questions 4 and 5 be statistically correlated to reliably predict potentially catastrophic malicious BGP routing instabilities while minimizing spurious alerts?
- Subsidiary research question #9: How can the new instability prediction model developed in this thesis be utilized to proactively protect BGP implementations against malicious catastrophic failure?
- Subsidiary research question #10: What evaluation criteria will be used to ensure that this model performs at or above an acceptable level of performance?

E. OUTLINE OF THESIS

The rest of this thesis will adhere to the following outline.

- Chapter II Background
- Chapter III Development of the BGP Instability Prediction Model
- Chapter IV Verifying the BGP Instability Prediction Model
- Chapter V Conclusions

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. BGP OVERVIEW

The Border Gateway Protocol (BGP) is an inter-autonomous system routing protocol. An autonomous system is a network or group of networks under a common administration and with common routing policies. BGP is used to exchange routing information on the Internet, and is the protocol used between Internet service providers (ISPs). BGP uses four different types of messages: open, update, keepalive, and notification.

An open message is used when a router running BGP tries to open a TCP connection (also known as a peering session) with another BGP router. An update message is used by a router: to withdraw destinations that have been advertised previously, to announce a route to a new destination, or both. Keepalive messages are exchanged regularly by peering routers to tell each other that they are alive. A router sends a notification message when either an error condition is detected or when a router wants to close the peering session.

B. BGP ROUTING INSTABILITY

1. Terms and Definitions

In this thesis, we are only interested in maliciously caused, catastrophic BGP routing instability events. Catastrophic BGP routing instability refers to exponential network performance degradation in both protocol and user levels. Exponential network performance degradation in protocol level refers to exponential or similarly fast growth in the rate of prefix updates, high update rates lasting for an extended time frame, with almost all prefixes churning, in BGP updates from almost all default-free peers. Exponential network performance degradation in user level refers to exponential or similarly fast growth in the rate of delay, hop-count, or reachability lasting for an extended time frame.

2. Causes

There are many causes of these instabilities; some are maliciously induced, while others are a natural occurrence of network operations. This thesis will focus on the maliciously induced instabilities.

3. Metrics for Measuring

Regardless of how the malicious instabilities are introduced, the effects on the targeted router can be measured by observing certain pertinent BGP routing metrics. These observed metrics can take two forms: micro- and macro-level.

Micro-level metrics are most closely associated with network layer (OSI layer 3) issues such as: number and distribution of prefixes that appear in the routing tables, their change over time, and the number of prefix advertisements and withdrawals in BGP update messages dispatched per unit time[1]. Macro-level metrics, on the other hand, are most closely associated with the end-user's perception of performance. Each of these metric forms has its strengths and limitations with respect to how reliable an indicator it is for a potential routing instability. We will further discuss these two forms of metrics and their strengths and limitations in chapter 2.

There are existing research reports on BGP routing instabilities. One of these reports is from Renesys Corporation, titled "Global Routing Instabilities Triggered by Code Red II and Nimda Worm Attacks." This research is mainly concerned with the micro-level measures of BGP routing instabilities. The metrics used in the Renesys report were: the amount of prefix-churning, the number of malformed updates caused by misconfiguration, and the intensity of CPU utilization. Their work was important but not sufficient. The micro-level metrics in their model seem to produce too many false-positives. Like the Renesys research, the AT&T research also resulted in an excess of false-positives.

This AT&T Labs research titled "BGP Routing Stability of Popular Destinations," also focused mainly on micro-level incidents. In this work, the metric used was the length of a measured "event," where the length of the event is determined by the number of updates for a given prefix.

A long event is usually an indication of prefix-churning or routing oscillation, whereas a short event usually indicates normal behavior. However, using the number of updates is not a good metric for determining the length of an event. For example, an event that consists of many short updates should not be considered to be a long event.

Most current models that attempt to predict a BGP routing instability rely exclusively on either macro- or micro-level metrics. Due to limitations inherent in each form of metric, there is an excessive rate of false positive and false negative warnings. To elaborate somewhat, consider the following. 1) Macro-level (user perception of performance) indications are by their nature too "late" to provide prior warning. 2) Macro-level indications may be too subtle to discern the early warnings that occur at the protocol level (e.g., Code Red and Nimda did not elicit user-level awareness). 3) Macro-level indications may warn of a catastrophic BGP performance problem unrelated to instabilities (e.g., a severed cable). 4) Micro-level indications could falsely suggest a catastrophic instability (false positive), when simple user perception (macro-level) might be sufficient to invalidate this.

C. BGP GLOBAL ROUTING INSTABILITY METRICS

So far, like any network performance evaluator, researchers can only use the existing performance metrics to evaluate their networks' performance. "Neither data nor even standard formats are available against which to compare performance with other networks or against some baseline. Nor are there reliable performance data for users to assess providers. Data characterization and traffic flow analysis are also virtually nonexistent, yet they remain essential to understanding the internal dynamics of the Internet infrastructure." [2]

In this chapter, we will introduce the two types of metrics used to measure BGP global routing instability: micro-level and macro-level metrics.

1. Micro-Level Metrics

a. Definition

Micro-level metrics are those used to measure network performance at the protocol level, where the emphasis is in specific points of the network.

b. Description

At the micro level, end-users cannot perceive any change in network performance at the protocol level. Examples of micro-level metrics are: number and distribution of prefixes that appear in the routing tables, their change over time, and the number of prefix advertisements and withdrawals in BGP update messages sent out per unit time[1]. Let's consider this. "The BGP protocol contains route flap dampening mechanism that prevents a BGP router from sending too many messages about an unstable route [43]; and a timer (the Minimum Route Advertisement Interval Timer) that maintains a minimum separation between consecutive announcements to a given peer, with default value of 30 seconds. Therefore, if we see large increases in the number of BGP update messages, it's an unambiguous sign that the diversity of network prefixes is rising." [1] Therefore, the number of BGP update messages is considered a micro-level metric because it gives out information about the network performance at the protocol level, but nothing about network performance at the user level.

c. Strengths and Limitations

Micro-level indications could falsely suggest a catastrophic instability (false positive), when there are not enough simultaneous micro-level performance failures to cause macro-level failures. In other words, users don't necessarily feel the degradation in network performance when micro-level network performance degradation exists. The strength of micro-level metrics is that, if used correctly in the network performance prediction model, it can give prior warning.

2. Macro-Level Metrics

a. Definition

Macro-level metrics are those used to measure network performance at the global, user level.

b. Description

These metrics are most closely associated with the end-user's perception of performance. They describe user traffic behavior. Examples of macro-level metrics are: hop counts, reachability/connectivity disruption, throughput, and response time.

c. Strengths and Limitations

Here are some of the limitations inherent in macro-level metrics: 1) By the time they are apparent to the user, the critical level of damage has already been reached. 2) Macro-level metrics may be too subtle to discern early warnings at the protocol-level. 3) Macro-level alerts may be out of scope of BGP routing instability (e.g., a severed cable). On the other hand, macro-level metrics offer the advantage of tangible, user awareness of catastrophic network failure.

THIS PAGE INTENTIONALLY LEFT BLANK

III. DEVELOPMENT OF THE BGP INSTABILITY PREDICTION MODEL

A. SELECTION OF METRICS FOR PREDICTION MODEL

During the course of selecting the appropriate metrics for the prediction models, the research shows that the round trip time (RTT) – a measurement of time delay between the time when a source sends out a packet and the time when the source receives an acknowledgement that the packet was indeed received by the intended destination – is a good candidate for the macro-level metric. Each sample RTT will be the average of all of the RTTs from all of the source-destination pairs during a fixed time interval. An equally good candidate for the micro-level metric would be network traffic created from the update messages. We will call this metric Update-Message Traffic. In an update message, a route from a fixed source to a new destination - as reflected in the prefix - is advertised to a neighbor destination. Another form of update message results from the withdrawal of routes associated with the previously advertised prefixes. Therefore, each sample Update-Message Traffic will be the total number of Update Messages obtained from all of the source / destination pairs during a fixed time interval, divided by that time interval. Here is how we use the macro- and micro-level metrics in our statistical model.

$x(i)$: user level performance metric value for sample interval i

$y(i)$: BGP routing instability metric for sa mple interval i

S : set of measured source-destination pair s in Skitter archive

R : set of data colle cting routers in Route View archive

$u(i, p)$: user-level metric value (e.g., RTT or hopcount) for source-destination pair p at sample interval i

$b(i, r)$: BGP metric value (e.g., number of U pdate messages r eceived) for router r at sample interval i

$$x(i) = \frac{\sum_{p \in S} u(i, p)}{\|S\|}$$

$$y(i) = \frac{\sum_{r \in R} b(i, r)}{\|R\|}$$

B. EXTRACTION OF PERTINENT METRICS FROM SUPPLIED DATA

The data necessary to develop and test the proposed model was obtained from the following, publicly available locations (URLs shown below, which were last accessed on March 09, 2004):

1. ftp://archive.routeviews.org/

This archive contains the historical BGP RIB (routing information base) table snapshots and routing BGP update messages collected by the University of Oregon's RouteViews project.

2. http://www.caida.org/

This site contains Skitter, a tool for actively probing the Internet in order to analyze topology and performance issues. It is maintained by CAIDA (Cooperative Association for Internet Data Analysis).

3. https://sk-data.caida.org:8444

This site maintains the Skitter Data Archive (SDA). It contains historical BGP macro-level routing information such as hop counts (NumHops), round trip time from source to final destination, and intermediate hop IP addresses.

Data relevant to the 2003 Blaster Worm and East Coast Blackout incidents are of specific interest to this thesis. There is a challenge in mining the desired information from these two, large data archives (the SDA has files that are at least 17MB each). Since the data bank is huge, we cannot extract it by hand, so scripts were written to filter the unwanted data. Here are the steps taken to extract and store the pertinent data:

1. Extracting BGP Update Messages

a. Download 15-Minute Data Files

Go to ftp://archive.routeviews.org/bgpdata/; click on the folder of the time period of interest (i.e. 2003.08 folder was what I chose for this thesis) and within this folder, choose UPDATES folder to download the Bzip2 files that contain BGP update messages collected during the time period of interest. For example, in the UPDATES folder, one will see Bz2 files, each consists of 15

minutes of BGP update information. Thus, there would be 24hrs/day*60min/hr*1file/15min=96 files/day. Each day (according to the name of the file), the first file will start at time 17:00:00 (approximately) of the previous day and since this kind of file captures 15 minutes of BGP update information, the next file will pick up at 17:15:00. For example, if we consider the file *updates.20030802.0007.bz2,* found in the UPDATES folder, one can see that this file is the first file of August 02, 2003. This file consists of BGP update information that starts from 17:00:07 (because of "*0007*" in the file name). So if one wants all of his/her files to start at the same time, 17:00:00, for data analysis, one then would also download the last file of August 01, 2003 because it may have captured BGP updates from 17:00:00 to 17:00:06 of August 02, 2003. The original data files are in the MRT format. "MRT software is in active use providing stress testing of commercial routers, collecting and analyzing Internet routing traffic for researchers, and serving as the primary routing software connecting networks to the Internet and the 6Bone."[4]

b. Decompress and Convert Data Files Into ASCII Format

First, decompress Bzip2 files using the command:

%Bunzip2 [bz2filename] > [output filename].

Then, go to http://www.mrtd.net/ to download and install the MRT-2.2.0a release. Two categories of tools exist within the MRT software package: Routing tools and Network Performance measurement tools. One of the tools in the package *Route_BtoA,* is used to convert MRT messages from binary to ASCII. The command used in the conversion process is

%route_btoa [MRT data file name] > [output ASCII file name]

Here is an example of a converted, ASCII file of MRT messages:

```
TIME: 07/31/03 17:03:17
TYPE: BGP4MP/MESSAGE/Update
FROM: 213.200.87.254 AS3257
TO: 198.32.162.102 AS6447
ORIGIN: IGP
ASPATH: 3257 3561 701 26501
NEXT_HOP: 213.200.87.254
MULTI_EXIT_DISC: 80
ANNOUNCE
  65.197.183.0/24
  65.219.31.0/24
```

```
TIME: 07/31/03 17:03:17
TYPE: BGP4MP/MESSAGE/Update
FROM: 147.28.255.1 AS3130
TO: 128.223.60.102 AS6447

TIME: 07/31/03 17:03:18
TYPE: BGP4MP/MESSAGE/Update
FROM: 196.7.106.245 AS2905
TO: 128.223.60.102 AS6447
WITHDRAW
  198.199.244.0/24
  198.199.245.0/24
  203.57.42.0/24
  207.191.23.0/24
```

c. Extracting Pertinent Information

Since the readable data files obtained from step b are large, approximately 4 MB each, a python script was written to filter the data and store only what was required for data analysis. See the python script "filter.py" in Appendix B. The fields that were retained from BGP update files were: date, time with military format and announcements of whether the update was an announce or a withdraw. Before running any script, one would have to change the mode to make the script executable by typing the command:

*%chmod +x [*scriptfilename*] (in this case, it is filter.py).*

To run the filter.py script, use the following command:

% python filter.py --um [infilename]>[outfilename].

The switch "*--um*" is a convention which means update messages. Since the original files were alphabetically ordered by date and time, it is wise to keep the file name the same, but with a different extension. "*.text*" or "*.dat*" is a good extension for data files. Sample output of a filtered BGP update file would be:

```
TIME: 02/11/04 17:00:00 WITHDRAW
TIME: 02/11/04 17:00:00 ANNOUNCE
TIME: 02/11/04 17:00:00 ANNOUNCE
TIME: 02/11/04 17:00:00 ANNOUNCE
TIME: 02/11/04 17:00:00 ANNOUNCE
TIME: 02/11/04 17:00:00 ANNOUNCE
```

d. Merge 15-Minute Files into One File for Whole Duration

I merged the 15-minutes files obtained from step c into one file for whole duration using the command "%cat *.text>[outfilename]". The meaning of this command is as follows. If, after extracting the files in step c, one saves his/her output files under ".text" extension, then, "*.text" represents all text files in the folder. Finally, all of these text files will be concatenated into one file with the new name. To prevent the output file from being overwritten, one would use ".dat" extension instead of the".text" extension, since "*.text" was previously used, therefore, if the output file is also a ".text" file, it will also be treated as one of the input files.

e. Compute 6-Second Samples

I used the data file obtained from step d to generate another file that consisted of the total number of updates for every 6 seconds. Since the data files obtained from step c are alphabetically ordered by date and time, the one file obtained from step d should be sorted internally by date and time. Therefore, it is not necessary to sort the file gotten from step d before starting this step e. A python script was written to compute 6-second samples. See the python script "timeslice.py" in Appendix B. Before running this script, one must change its mode to make it executable. To run the timeslice.py script, use the following command:

% python timeslice.py --um [infilename]>[outfilename].

See the sample output of the file of this type in Appendix A, which shows the first page and the last page of the file "TotalUpdates_6sec.dat".

2. Extracting User Level Data

a. Download 24-Hour Data Files from 9 Servers

Go to https://sk-data.caida.org:8444 (researchers desiring data from this archive are given login names and passwords to access the archive after first registering at http://www.caida.org/tools/measurement/skitter/skitter_request.xml). After signing onto this site, different folders are shown, each with a name that represents the system that monitors Internet traffic. To ensure a proper

sampling of global network events, network behavior from multiple source and destination pairs are considered. CAIDA currently maintains 19 Skitter hosts all over the world. However, not all Skitter monitors are running the full destination set at all times. Therefore, data for this study was taken from nine DNS servers (a-root, b-root, d-root, e-root, g-root, h-root, i-root, k-peer, and m-root) at the archive. These servers run the full destination set at all times. Folder "a-root" was chosen arbitrarily. After clicking on the folder "a-root", click on the folder of the year of interest, and finally click on the folder of the month of interest. At this point, the zipped ARTS files are visible with gzip extension. "ARTS is a binary file format specification for storing network data. CAIDA distributes some simple applications for viewing and manipulating ARTS data. The entire package is called arts++."[5]

b. Decompress and Covert to ASCII

Decompress the gzip files by using the command:

%gunzip [somefilename.gz].

Decompressing the gzip files yields arts files. Before converting the arts files into ASCII files, the arts++ tool package must be downloaded. Go to ftp://ftp.caida.org/pub/arts++/ to download the arts++ software package. It has several versions. For this thesis, I chose to use arts++-1-1-a9 version. After downloading, unzipping, and installing the arts++ package, type the following command to convert the arts files into ASCII files:

%artsdump [artsfilename] > [asciifilename].

There were some problems compiling the arts++ that are worth noting. The arts++ version that I downloaded did not compile under GCC 3.2.3 compiler of my Linux machine. Therefore, I had to tweak the code from the arts++ to make it work and hopefully this change also supports other older version(s) of GCC. Unfortunately, I don't have (easy) access to an older GCC version, which is what prompted the porting in the first place. The changes are all relatively straightforward. They consist of exposing the code to the std namespace where appropriate, including the iterator header where appropriate, properly casting source types when reading or writing binary data using streams,

and not duplicating default arguments in function definitions when they are already made explicit in the corresponding function declarations. Other details can be found in the diff. It may be necessary to macro-select the appropriate code based on the version of GCC (particularly the iterator header). My changes were made against the 1-1-a9 release, and the diff was made from the parent directory of the release in unified format, so it can be patched from within the release directory with:

%patch -p1 <../arts++-1-1-a9.gcc-3.diff

GCC 3 still issues a few warnings, but they didn't prevent arts++ from compiling. One can find the diff file pasted in Appendix E. Sample of an arts file after converting it into a readable text file would look like the following:

```
HEADER
        magic: 57264 (0xdfb0)
        identifier: 12288 (0x3000)
        version: 0 (0x0)
        flags: 0 (0x0)
        num_attributes: 1 (0x1)
        attr_length: 12 (0xc)
        data_length: 53 (0x35)
ATTRIBUTE
        creation: 06/30/2001 17:00:07 (0x3b3e6807)
IPPATH OBJECT DATA
        Src: 216.168.227.250  (0xfae3a8d8)
        Dst: 209.8.51.132     (0x843308d1)
        Rtt: 0 ms
        HopDistance: 0 (0x0)
        IsComplete: false
        NumHops: 7 (0x7)
                HopNum:   1 IpAddr: 216.168.227.1   (0x1e3a8d8)
                HopNum:   2 IpAddr: 157.130.32.241  (0xf120829d)
                HopNum:   3 IpAddr: 152.63.37.42    (0x2a253f98)
                HopNum:   4 IpAddr: 146.188.162.241 (0xf1a2bc92)
                HopNum:   5 IpAddr: 152.63.35.201   (0xc9233f98)
                HopNum:   6 IpAddr: 157.130.46.14   (0xe2e829d)
                HopNum:   7 IpAddr: 64.50.135.182   (0xb6873240)

HEADER
        magic: 57264 (0xdfb0)
        identifier: 12288 (0x3000)
        version: 0 (0x0)
        flags: 0 (0x0)
        num_attributes: 1 (0x1)
        attr_length: 12 (0xc)
        data_length: 78 (0x4e)
ATTRIBUTE
        creation: 06/30/2001 17:00:07 (0x3b3e6807)
IPPATH OBJECT DATA
        Src: 216.168.227.250  (0xfae3a8d8)
        Dst: 199.38.51.132    (0x843326c7)
        Rtt: 54.266 ms
        HopDistance: 14 (0xe)
        IsComplete: true
        NumHops: 12 (0xc)
```

```
HopNum:   1 IpAddr: 216.168.227.1   (0x1e3a8d8)
HopNum:   2 IpAddr: 157.130.32.241  (0xf120829d)
HopNum:   3 IpAddr: 152.63.37.58    (0x3a253f98)
HopNum:   4 IpAddr: 146.188.162.253 (0xfda2bc92)
HopNum:   5 IpAddr: 152.63.35.193   (0xc1233f98)
HopNum:   6 IpAddr: 192.205.32.133  (0x8520cdc0)
HopNum:   7 IpAddr: 12.123.9.50     (0x32097b0c)
HopNum:   8 IpAddr: 12.122.2.82     (0x52027a0c)
HopNum:   9 IpAddr: 12.122.1.206    (0xce017a0c)
HopNum:  10 IpAddr: 12.122.2.209    (0xd1027a0c)
HopNum:  11 IpAddr: 12.122.2.214    (0xd6027a0c)
HopNum:  12 IpAddr: 12.125.72.6     (0x6487d0c)
```

c. Extract Pertinent Information

Since the readable data files obtained from step b are so large, (approximately a 500-600 MB each) a python script was written to filter the data and to store what was required for data analysis. See the python script “filter.py” in Appendix B. This is the same script we used to extract the BGP update files with the switch --rtt instead of --um. For files with user level data, we filter for the date, with military format and round trip time (RTT) in milliseconds. For a quick reference, we now call a file with user level data RTT file. To run the filter.py script, use the following command:

% python filter.py --rtt [infilename]>[outfilename].

The mode of the script must be changed to make it executable. The switch “--rtt” is a convention which means round trip time. Since the original files were alphabetically ordered by date, time, and network traffic monitor names (9 dns severs), it is wise to keep the file name the same, but with different extension. “.text” or “.dat” is a good extension for data files. Sample output of a filtered RTT file would be:

```
creation: 08/27/2003 17:00:01 (0x3f4d4601) Rtt: 188.95 ms

creation: 08/27/2003 17:00:01 (0x3f4d4601) Rtt: 183.788 ms
creation: 08/27/2003 17:00:01 (0x3f4d4601) Rtt: 251.404 ms
creation: 08/27/2003 17:00:01 (0x3f4d4601) Rtt: 165.76 ms
```

d. Merge 24-hour Files into One File for Whole Duration and All Servers

The 24-hour files from all servers for the whole duration gotten from step c are merged into one file by using the command:

%cat *.text>*[outfilename].*

e. Sort the Data Based on Time

Since the files gotten from step c were alphabetically ordered by file name, when concatenated into one file in step d, this file is not sorted by date and time. We have to sort this file before going to step f by using the command:

%sort –n [unsortedfilename]>[sortedtfilename].

f. Compute 6-Second Samples

The data file obtained from step e is used to generate another file that consists of the average RTT of all of the source/destination pairs for each 6 second interval. A python script was written to do the computation. See the python script "timeslice.py" in Appendix B. This is the same script that we used to compute the 6-second samples for BGP updates. To run the timeslice.py script, use the following command:

% python timeslice.py --rtt --ave --slice=n [infilename]>[outfilename].

n is the time slice in seconds. For this thesis, n=6, but n may be any integer starting from 1. Note that n should not be too much larger than 6 because for a fast network, 6 seconds mean a lot of traffic. Therefore, large time slices can cause inaccuracy in data analysis. We can apply this same script to compute the median RTT of all of the source/destination pairs for every 6 seconds by using the command:

% python timeslice.py --rtt --med --slice=n [infilename]>[outfilename].

See the sample output of the file of this type in Appendix A. In Appendix A, one will see the first page and the last page of the files "RTT_ave_6sec.dat" and "RTT_median_6sec.dat".

3. Difficulties We Encountered During Data Gathering Process

First, each BGP update file captures fifteen-minute-interval update information. This yields 96 BGP update files for every day sampled. Secondly, the file names aren't intuitive. For example, the update file name "updates.20030810.0024" is a collection of 15 minutes of update information beginning at 17:00:24 on August 9, 2003. The first eight digits of the file name represent the date of the captured information, using the format: yyyymmdd. If the day field of a file name is 10; then the data inside this file can be of

information of any time between 17:00:00 of the 9th to 16:59:59 of the 10th. In other words, the day of captured information starts at 17:00:00 of the previous day and goes on for 24 hours.

C. NEW STATISTICAL MODEL

1. Micro- and Macro-Level Combination

This model will have two different variables: one, chosen from the micro-level metrics and the other, chosen from the macro-level metrics. Their periodic measurements can be considered as samples from two discrete-time, random processes. Then, the two variables will be tested for cross correlation.

2. Correlation Analysis

We want to use cross correlation analysis to test two random processes and to determine whether there is a cause-effect relationship between the two variables after a time delay. Cross correlation is a good fit for this thesis because it can be conjectured that when some BGP misbehavior causes a global routing problem, there is some delay between the two events. Cross correlation is defined as "the expected value of the product of a random variable from one random process with a time-shifted, random variable from a different random process."[3] The general formula for a discrete cross correlation of two discrete-time random processes would be $Rxy(\tau)=E\{x(t)y(t-\tau)\}$ with τ being all possible delays. We can then reduce the formula to $Rxy(k\Delta T)=1/(N+1)(\sum_{i=0}^{N-k} x(i)y(i+k))$ where ΔT is our sampling interval, say, 15 minutes; N is the number of sample tests; k is an integer that represents the sample number that we are investigating. For example, if we have N=10 (1..10), then k can be any integer running from 1 to 10; so if k=2, then $k\Delta T$ will tell us that we are investigating the second sample test, which starts at time t=15 since the first sample test would start at time t=0. So, for every $k_i\Delta T$ (i={0..N-k}), we will have a corresponding $Rxy(k_i\Delta T)$. With all the $Rxy(k_i\Delta T)$ being calculated, we will then choose the maximum of ($Rxy(k_0\Delta T)$… $Rxy(k_{N-k}\Delta T)$); we will call this chosen max value R. Say, R= $Rxy(k_5\Delta T)$ and ΔT=15 minutes. We then have the value $k_5\Delta T$=75 minutes, which can be used in our prediction model. The case would be that, for

example, at 2:00 A.M we experience some catastrophic network performing failure at the micro level; we can predict that at 3:15 A.M (75 minutes after 2:00 A.M) there may be some resultant catastrophic network performing failures happening at the macro level. We also are interested in the following properties of cross correlation:

a. If $E\{x(t)y(t-\tau)\}=E[X(t)]E[Y(t-\tau)]$, then X and Y are independent and uncorrelated.
b. $Rxy(\tau)\neq Ryx(\tau)$ (in our case, macro-level failures do not cause micro-level failures in network performance)
c. $Rxy(\tau)=Ryx(-\tau)$

THIS PAGE INTENTIONALLY LEFT BLANK

IV. VERIFYING THE BGP INSTABILITY PREDICTION MODEL

A. TIMELINESS

This model would help to calculate the time it takes for a catastrophic event to happen at the macro-level (user-level) after catastrophic network performance degradation at the micro-level (protocol-level). This gives a system administrator, or whomever is in charge of network performance sufficient time to react and thus to prevent the catastrophic event at the user-level.

B. RELIABILITY

As discussed previously, most current models that attempt to predict a BGP routing instability rely exclusively on either macro- or micro-level metrics, thus yielding an unsatisfactory rate of false positive and false negative warnings. This prediction model is an improved version of BGP instability prediction model, because it statistically combines both metric forms. In particular, the model will cue on two events occurring simultaneously as an indicator of an impending catastrophic instability; the exponential degradation of the chosen performance metrics, and a sufficiently strong correlation between both macro- and micro-level metrics over an extended interval.

1. Reduced False Negatives

This model gives fewer false negatives than existing models. It does not falsely predict that there is a catastrophic degradation in network performance based only on degradation in network performance at either micro-level or macro-level alone, where there is no observed effect on the users.

2. Reduced False Positives

This model gives fewer false positives than existing models, since it takes into account the correlation between interval-related macro- and micro-level observations of degradation. In other words, this model does not falsely predict a normal state when a macro-level, catastrophic event follows catastrophic network performance degradation at the micro-level (protocol-level).

C. DATA ANALYSIS

As mentioned earlier, the periods of interest for data analysis were the Blaster Worm period, specifically August 11, 2003, and the East Coast Blackout period, which occurred between August 14 and 15, 2003. To ensure that we covered the events completely, we gathered data for the interval from 17:00:00 of August 10, 2003 to 16:59:59 of August 12, 2003 for the Blaster Worm event. For the East Coast Blackout event, we studied the data that ranges from 17:00:00 of August 13, 2003 to 16:59:59 of August 16, 2003. In addition to the Blaster Worm and the East Coast Blackout events, we also studied the data from 17:00:00 of February 11, 2004 to 15: 59:59 of February 14, 2004. The reason for collecting data for this period is because the East Coast Blackout event (August 13-16, 2003) happened on Wednesday through Saturday of the week and February 11-14, 2004 was the most current "Wednesday through Saturday" period during the time the study was done. Data was compared between these to periods to normalize perturbations endemic to the day(s) of the week.

From these three periods of interest, three kinds of files were created: Total updates for every 6 seconds ("TotalUpdates_6sec.dat"), Average RTT of all source/destination pairs for every 6 seconds ("RTT_ave_6sec.dat"), and Median RTT of all source/destination pairs for every 6 seconds ("RTT_med_6sec.dat"). Please refer to the Data Collection section of this thesis to learn how those three kinds of files were produced. See Appendix A for the examples of the files' outputs.

We correlated the total updates column from TotalUpdates_6sec.dat file with the median RTT column of the RTT_med_6sec.dat file (or the average RTT column of the RTT_ave_6sec.dat file). In this thesis, we correlated both total updates with RTT medians and total updates with RTT averages.

We use Exponential Moving Average (EMA) to smooth the data series by using an average of the data, thus making it easier for us to spot trends and to reduce the influence of minor anomalies.

EMAs reduce lag by applying more weight to recent sample values than to older sample values. The weighting applied to the most recent sample depends on the length of the moving average interval. The shorter the exponential moving average interval, the more weight applied to the most recent sample. Software such as Matlab supports EMA calculation.

Before correlating the RTT (either median or average) with the total updates from the common period of interest, we calculate the zero-averaged RTT and zero-averaged total updates. Then, we used Matlab to find EMAs for both of the zero-averaged RTT and zero-averaged total updates. We calculated the EMAs with different window sizes (1, 2, 8, 32, 128, 512, 2048, and 8192). Inspecting the correlation graphs of the RTT and total update data before applying EMA disclosed that the early portion of the graphs was considerably smooth and normal. Therefore, we decided to truncate the data that is not useful for correlation, as it is likely to skew the correlation data, or may cause artifact to the correlation process. As an example, if the EMA RTT has 1000 samples in its data list and the window size used in EMA calculation is 32, then the RTT used for correlation is an array of sample points made up of the EMA RTT's 33^{rd} to 1000^{th} elements. Finally, the correlation process will normalize RTT and total update values so that the auto-correlations at zero lag are identically 1.0. On the next page, you will find the graphs of the EMA RTT and EMA total updates from the three periods of interest discussed earlier. Figure 1, Figure 2 and Figure 3 show the plots of EMA RTT medians and EMA total updates of the three periods of interest, all with a moving window size of 512 samples. See Appendix C for the plots of EMA RTT averages and EMA total updates of the three periods of interest. As evinced from the three figures, the BGP update count trend shows variation, with periodic, transient peaks.

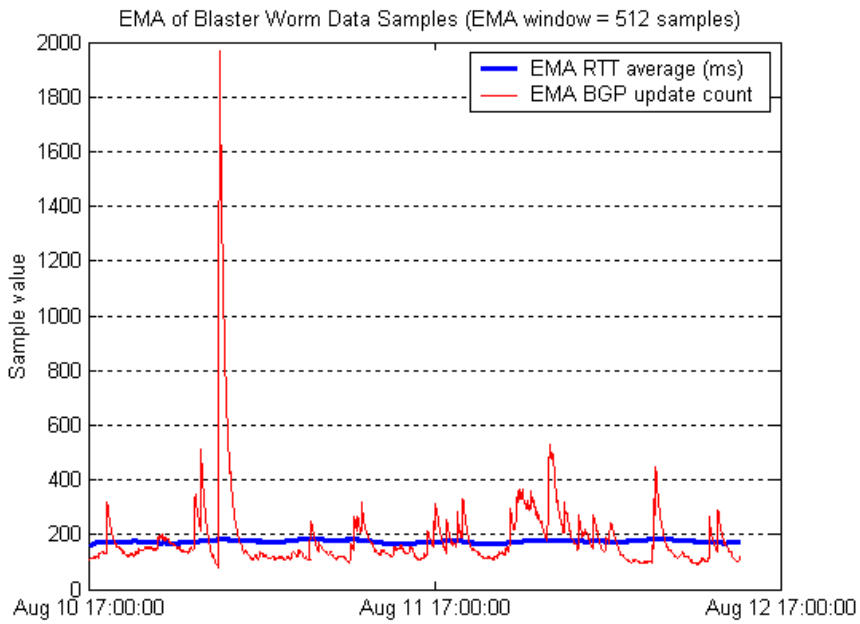

Figure 1. Blaster Worm Data

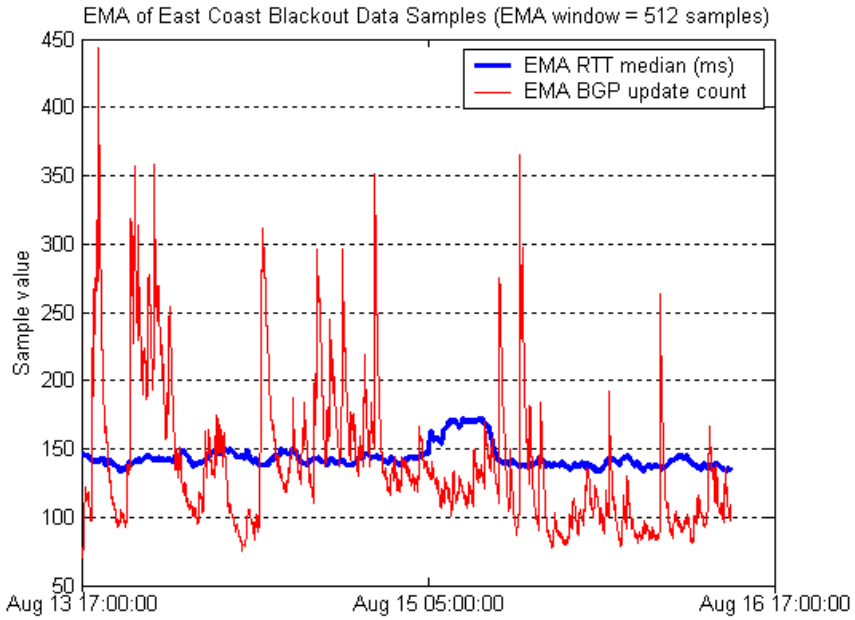

Figure 2. East Coast Blackout Data

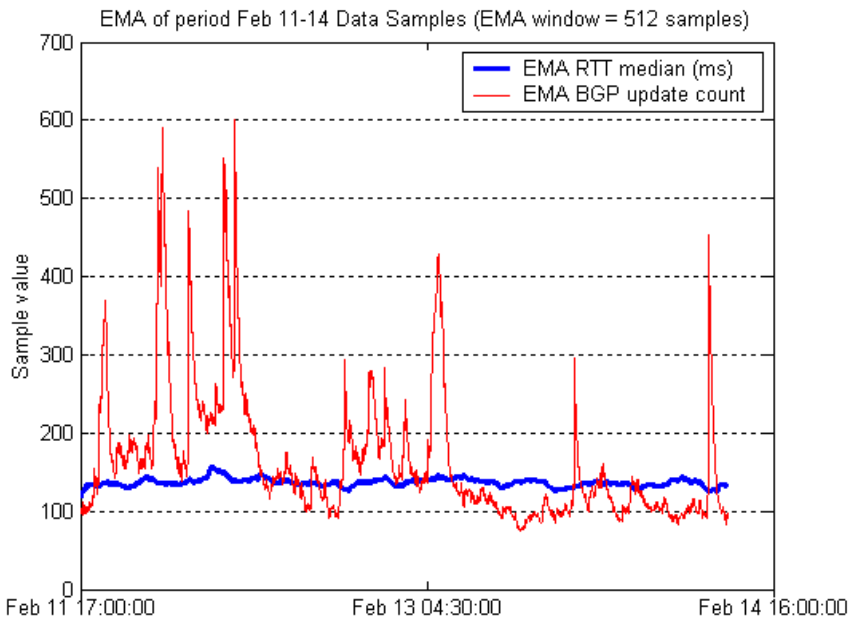

Figure 3. Feb 11-14

On the other hand, the RTT traffic plot is relatively smooth, with the exception of one obvious bump in the East Coast Blackout graph, occurring around 5:00:00 on August 15, 2003 (Figure 2. East Coast Blackout data). From careful observation of the graphs, the total update traffic trend and the RTT traffic trend don't seem to correlate because, despite the spikes in the total updates plot, the RTT seems normal. Since there is a bump in the RTT of the East Coast Blackout (Figure 2), we hope to see some correlation in the RTT and total updates for this period. Figure 4 shows the results of the correlation of RTT medians and total updates for the East Coast Blackout period with various EMA moving window sizes (see Appendix C for the Matlab outputs of the max correlation coefficient and its lag and min correlation coefficient and its lag for various EMA window sizes of all three periods: East Coast Blackout, Blaster Worm and February 11-13). We chose to show just the correlation of the total updates with RTT medians instead of also including the RTT averages because the results don't vary substantially. However, we appended the graphs of correlation of the total updates with RTT averages and the corresponding Matlab outputs of max correlation coefficient, min coefficient and their lag values in

Appendix C. The y-axis represents correlation coefficients and the x-axis represents time lags. Therefore, a point on the graph can tell us the extent to which the RTT and total updates correlate for a certain time lag value. In the correlation analysis section of this thesis, we have discussed that the closer the absolute value of the correlation coefficient is to 1, the more correlated are the random variables (in this case RTT and total updates). Furthermore, if the coefficient is zero or close to zero, the variables are considered to be uncorrelated. In this thesis, we are looking for the trend of strong correlation between the RTT and the total updates. We feel that any coefficient lower than 0.5 indicates low correlation. Moreover, at time lag value of zero, it doesn't matter how close the coefficient is to 1, it shows that the RTT and the total updates are not correlated at that point where time lag is equal to zero. From figure 4, one can see how the data are smoothed as window size is increased from 1.

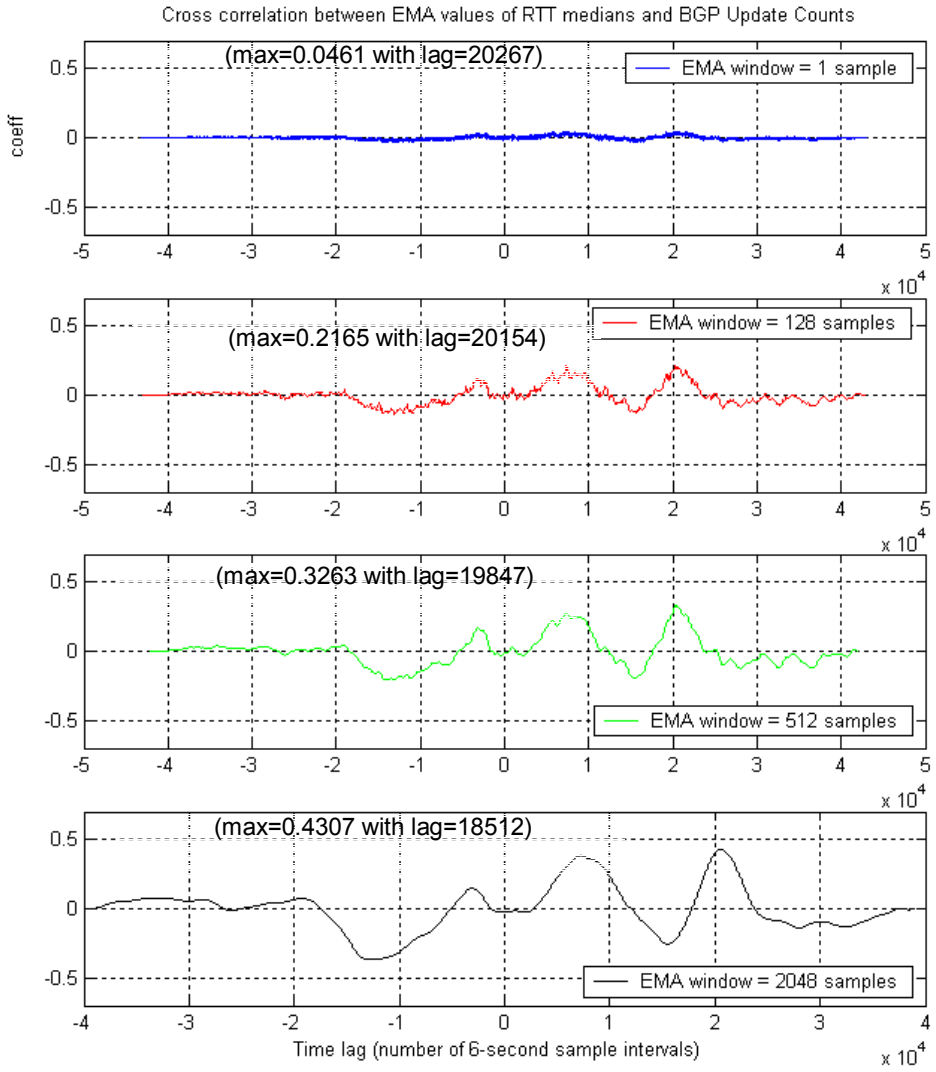

Figure 4. Correlation of RTT and total updates for the East Coast Blackout period

We should be aware that choosing the correct window size for correlation is very important since the shorter the exponential moving average (EMA window size), the greater the weight applied to the most recent sample value. Therefore, we don't want the window size to be so small that the historical data has very little effect on the EMA. Conversely, too great a window size unduly reduces the importance of the most recent data in terms of its effect on EMA. A casual glance at Figures 4, 5, and 6, might seem to imply that the RTT and the total updates are strongly correlated if the peaks are higher than 0.5. However, note that as

window size increases, so do the peaks. Since our data is comprised of 6-second samples, the moving window size of 512 is a good choice. This assertion is substantiated because we have 512 samples * 6 sec/sample = 3072 seconds (51.2 minutes) worth of network traffic data. On the other hand, if we choose a window size of 2048, we would do the EMA on 204.8 minutes (3.41 hours) of network traffic data. This means that an event occurring 3 hours earlier than time-zero, with a strong contribution to the bad event that is expected to occur at time-zero will be assigned a small weight in the calculation of the current EMA value. This may adversely affect our ability to spot the trend which predicts the catastrophic event.

Contrary to expectations, according to figure 4, the subplot whose window is 512 samples show no strong correlation between the RTT medians and the BGP update counts. Similarly, we can see that neither the figure 5 subplot nor that from figure 6 shows a strong correlation between the RTT medians and the BGP update counts. Most of the coefficients are below 0.5 however, for the coefficients that are around 0.5, their corresponded time lag values approach zero. This also provides evidence of weak correlation. The tests show no strong correlation between the RTT (user level metric) and the BGP update count (protocol level metric).

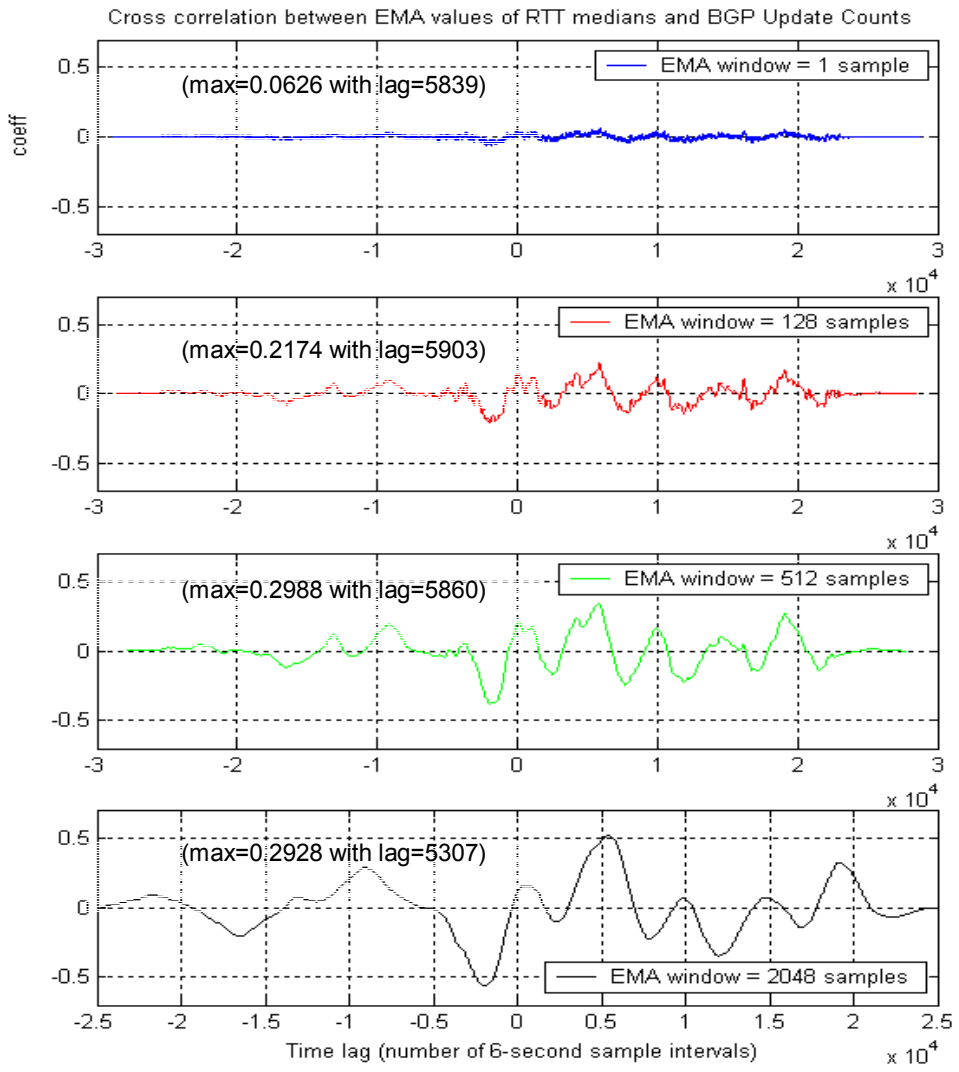

Figure 5. Correlation of RTT and total updates of the Blaster Worm period

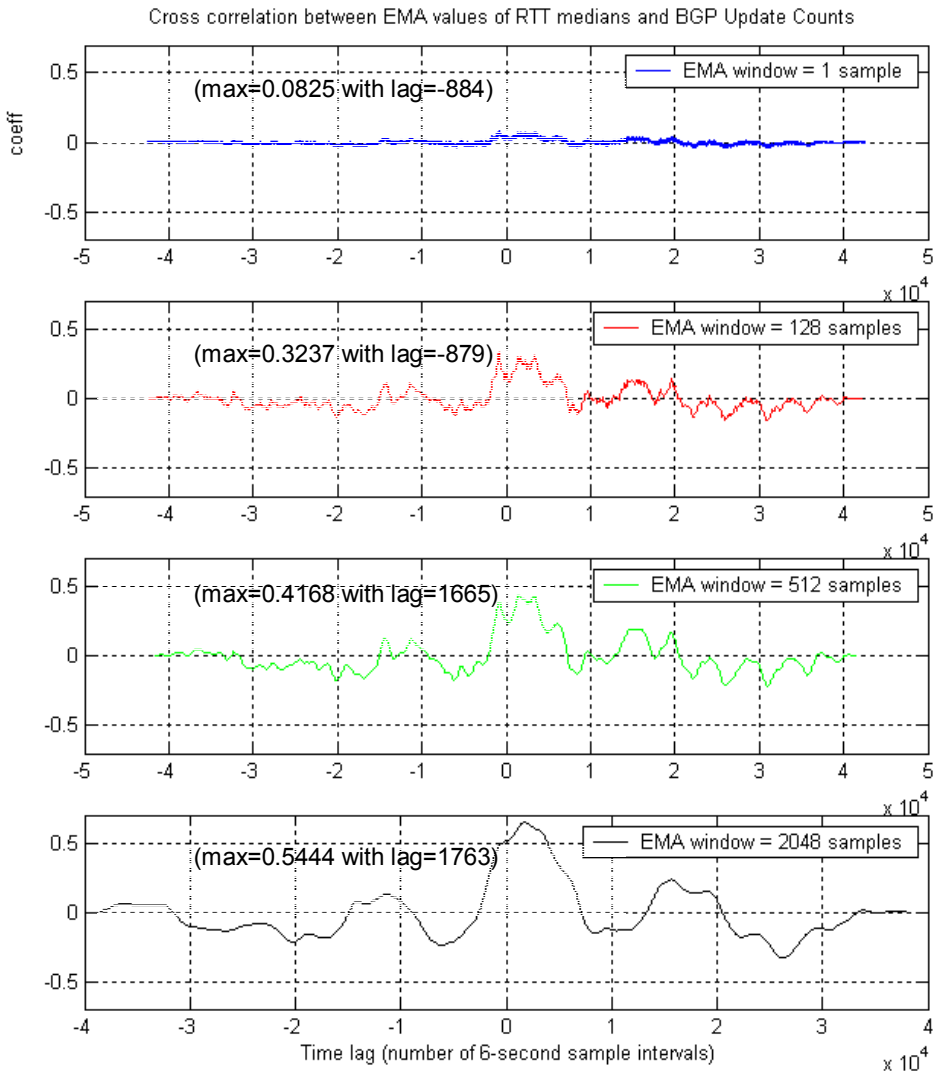

Figure 6. Correlation of RTT and total updates of the Feb 11-14 period

V. CONCLUSIONS

A. SUMMARY OF MAIN CONTRIBUTIONS

It was the original intent of this thesis to develop an improved BGP instability prediction model by statistically combining BGP instability metrics with user level performance metrics. The motivation for such a model is twofold. 1) To provide sufficient prior warning of impending failure to facilitate proactive protection measures. 2) To improve warning reliability beyond existing models, by demonstrably reducing both false positives and false negatives. However, based on our analysis of actual network trace data, it shows that a widely used BGP instability metric, the total number of update messages received in a time period, is not a good indicator of future user level performance.

B. LESSONS LEARNED

From different tests and correlations of the user level metric (RTT) and the protocol level metric (BGP update counts), we found no evidence of strong correlation of the metrics. The conclusion is that the metrics are stochastic and without significant correlation. Although the ideal conclusion of any thesis may be the substantiation of the main hypothesis, it is just as valuable – as in this case – when the hypothetical contention is demonstrated to be without basis.

C. RECOMMENDATIONS FOR FURTHER RESEARCH

Although no evidence of strong correlation of the metrics was found, there is no strong evidence that shows that the metrics are indeed uncorrelated. It is recommended for the future researchers to do the following:

- If possible, define what the model of an attack should look like for both at the protocol and user levels. If one is talking exclusively about maliciously induced instabilities such as those caused by worm injections, then it is recommended that one should define what the model of a certain kind of worm attack should look like at both the protocol and user levels. It is important for one to collect data and perform studies on different worm attacks to obtain

consistent patterns between different worm attacks if that is the case.

- Once the patterns for the attack are developed for the protocol and user levels (one may want to build a model signal for each attack pattern developed), one can correlate the protocol level metric with its corresponding model signal. Similar correlation should be done for the user level metric and its corresponding model signal. Finally, one could calculate the lag between the protocol level event and the user level event. Perhaps the result from the protocol level correlation will show that there is an attack signature found at the protocol level at 17:00:00, and perhaps the result from user level correlation will show that there is an attack signature found at the user level at 19:00:00. In such an instance, we may then be able to say that the lag is 2 hours.

APPENDIX A DATA USED IN TESTING THE PREDICTION MODEL

The following is the first page and the last page of the TotalUpdates_6sec.dat mentioned in the Data Analysis section. This shows actual data gotten from the Blaster Worm period.

Second of Day	Date	Number of Updates
61200	08/10/03	106
61206	08/10/03	357
61212	08/10/03	43
61218	08/10/03	82
61224	08/10/03	109
61230	08/10/03	236
61236	08/10/03	142
61242	08/10/03	53
61248	08/10/03	142
61254	08/10/03	222
61260	08/10/03	301
61266	08/10/03	145
61272	08/10/03	64
61278	08/10/03	95
61284	08/10/03	298
61290	08/10/03	73
61296	08/10/03	110
61302	08/10/03	70
61308	08/10/03	71
61314	08/10/03	181
61320	08/10/03	190
61326	08/10/03	26
61332	08/10/03	78
61338	08/10/03	62
61344	08/10/03	252
61350	08/10/03	120
61356	08/10/03	18
61362	08/10/03	76
61368	08/10/03	104
61374	08/10/03	222
61380	08/10/03	95
61386	08/10/03	11
61392	08/10/03	58
61398	08/10/03	187
61404	08/10/03	189
61410	08/10/03	28
61416	08/10/03	28
61422	08/10/03	64
61428	08/10/03	147
61434	08/10/03	266
61440	08/10/03	28
61446	08/10/03	73

.....

.....

.....

60876	08/12/03	97
60882	08/12/03	113
60888	08/12/03	98
60894	08/12/03	63
60900	08/12/03	152
60906	08/12/03	131
60912	08/12/03	184
60918	08/12/03	26
60924	08/12/03	144
60930	08/12/03	184
60936	08/12/03	137
60942	08/12/03	75
60948	08/12/03	47
60954	08/12/03	63
60960	08/12/03	75
60966	08/12/03	165
60972	08/12/03	47
60978	08/12/03	59
60984	08/12/03	271
60990	08/12/03	137
60996	08/12/03	113
61002	08/12/03	98
61008	08/12/03	132
61014	08/12/03	117
61020	08/12/03	110
61026	08/12/03	105
61032	08/12/03	75
61038	08/12/03	2
61044	08/12/03	177
61050	08/12/03	108
61056	08/12/03	45
61062	08/12/03	67
61068	08/12/03	70
61074	08/12/03	49
61080	08/12/03	73
61086	08/12/03	89
61092	08/12/03	39
61098	08/12/03	25
61104	08/12/03	148
61110	08/12/03	17
61116	08/12/03	61
61122	08/12/03	147
61128	08/12/03	82
61134	08/12/03	76
61140	08/12/03	57
61146	08/12/03	62
61152	08/12/03	84
61158	08/12/03	19
61164	08/12/03	159

61170	08/12/03	79
61176	08/12/03	72
61182	08/12/03	144
61188	08/12/03	136

The following is the first page and the last page of the RTT_median_6sec.dat mentioned in the Data Analysis section. This shows actual data gotten from the Blaster Worm period.

Second of Day (ms)	Date	Number of Src/Dest pairs	Median RTT
61200	08/10/2003	195	106.293
61206	08/10/2003	196	130.864
61212	08/10/2003	210	117.376
61218	08/10/2003	217	124.917
61224	08/10/2003	196	117.576
61230	08/10/2003	194	127.079
61236	08/10/2003	203	129.37
61242	08/10/2003	219	133.75
61248	08/10/2003	223	137.858
61254	08/10/2003	175	135.05
61260	08/10/2003	187	138.275
61266	08/10/2003	208	114.332
61272	08/10/2003	165	133.852
61278	08/10/2003	202	136.458
61284	08/10/2003	227	142.93
61290	08/10/2003	230	142.905
61296	08/10/2003	226	137.289
61302	08/10/2003	225	152.76
61308	08/10/2003	144	149.994
61314	08/10/2003	195	110.322
61320	08/10/2003	200	135.736
61326	08/10/2003	188	123.34
61332	08/10/2003	200	119.12
61338	08/10/2003	187	145.409
61344	08/10/2003	224	130.923
61350	08/10/2003	199	140.457
61356	08/10/2003	236	140.92
61362	08/10/2003	199	137.662
61368	08/10/2003	185	112.761
61374	08/10/2003	185	127.458
61380	08/10/2003	223	137.159
61386	08/10/2003	223	142.152
61392	08/10/2003	208	126.866
61398	08/10/2003	191	141.587
61404	08/10/2003	198	119.181
61410	08/10/2003	205	138.066
61416	08/10/2003	204	147.028
61422	08/10/2003	174	131.663
61428	08/10/2003	173	134.512
61434	08/10/2003	198	139.725
61440	08/10/2003	201	135.819
61446	08/10/2003	225	161.826
61452	08/10/2003	185	115.794
61458	08/10/2003	179	158.968

```
61464           08/10/2003  160                     142.654
61470           08/10/2003  201                     138.615
61476           08/10/2003  219                     135.032
61482           08/10/2003  175                     132.97
61488           08/10/2003  174                     146.896

...

...
60888           08/12/2003  227                     133.987
60894           08/12/2003  208                     137.869
60900           08/12/2003  202                     146.078
60906           08/12/2003  225                     145.43
60912           08/12/2003  201                     145.848
60918           08/12/2003  211                     128.911
60924           08/12/2003  198                     162.274
60930           08/12/2003  239                     149.152
60936           08/12/2003  222                     131.228
60942           08/12/2003  229                     144.681
60948           08/12/2003  180                     133.966
60954           08/12/2003  210                     142.857
60960           08/12/2003  221                     160.725
60966           08/12/2003  215                     134.751
60972           08/12/2003  218                     157.365
60978           08/12/2003  229                     151.365
60984           08/12/2003  264                     143.096
60990           08/12/2003  216                     152.404
60996           08/12/2003  227                     152.851
61002           08/12/2003  203                     143.659
61008           08/12/2003  232                     156.745
61014           08/12/2003  216                     153.438
61020           08/12/2003  208                     136.858
61026           08/12/2003  205                     142.604
61032           08/12/2003  224                     141.911
61038           08/12/2003  238                     151.396
61044           08/12/2003  210                     166.871
61050           08/12/2003  196                     139.584
61056           08/12/2003  205                     151.676
61062           08/12/2003  206                     157.095
61068           08/12/2003  222                     137.303
61074           08/12/2003  196                     118.621
61080           08/12/2003  217                     139.841
61086           08/12/2003  226                     157.672
61092           08/12/2003  198                     130.577
61098           08/12/2003  201                     146.625
61104           08/12/2003  219                     134.969
61110           08/12/2003  221                     156.115
61116           08/12/2003  207                     148.629
61122           08/12/2003  205                     136.413
61128           08/12/2003  195                     144.452
61134           08/12/2003  219                     152.147
61140           08/12/2003  201                     151.559
61146           08/12/2003  207                     122.02
61152           08/12/2003  208                     139.373
61158           08/12/2003  247                     135.362
```

61164	08/12/2003	195	138.961
61170	08/12/2003	186	141.083
61176	08/12/2003	197	127.375
61182	08/12/2003	212	137.193
61188	08/12/2003	204	140.467

The following is the first page and the last page of the RTT_avg_6sec.dat mentioned in the Data Analysis section. This shows actual data gotten from the Blaster Worm period.

Second of Day	Date	Number of Src/Dest pairs	Average RTT (ms)
61200	08/10/2003	195	150.430317949
61206	08/10/2003	196	166.040071429
61212	08/10/2003	210	142.786619048
61218	08/10/2003	217	159.889820276
61224	08/10/2003	196	165.982362245
61230	08/10/2003	194	160.669226804
61236	08/10/2003	203	165.920778325
61242	08/10/2003	219	165.858
61248	08/10/2003	223	181.005865471
61254	08/10/2003	175	162.420628571
61260	08/10/2003	187	172.748973262
61266	08/10/2003	208	148.718572115
61272	08/10/2003	165	160.320254545
61278	08/10/2003	202	164.797430693
61284	08/10/2003	227	165.139198238
61290	08/10/2003	230	167.179465217
61296	08/10/2003	226	172.506154867
61302	08/10/2003	225	168.950511111
61308	08/10/2003	144	178.9145625
61314	08/10/2003	195	154.965394872
61320	08/10/2003	200	167.97936
61326	08/10/2003	188	162.540361702
61332	08/10/2003	200	160.249595
61338	08/10/2003	187	177.294304813
61344	08/10/2003	224	157.555745536
61350	08/10/2003	199	160.626613065
61356	08/10/2003	236	167.269266949
61362	08/10/2003	199	150.198361809
61368	08/10/2003	185	176.121427027
61374	08/10/2003	185	168.079545946
61380	08/10/2003	223	161.399269058
61386	08/10/2003	223	180.654829596
61392	08/10/2003	208	162.067052885
61398	08/10/2003	191	171.993172775
61404	08/10/2003	198	150.439762626
61410	08/10/2003	205	170.116882927
61416	08/10/2003	204	173.172833333
61422	08/10/2003	174	159.721954023
61428	08/10/2003	173	165.566901734
61434	08/10/2003	198	174.01089899
61440	08/10/2003	201	169.199482587
61446	08/10/2003	225	187.402395556
61452	08/10/2003	185	161.321162162

61458	08/10/2003	179	179.899139665
61464	08/10/2003	160	170.55150625
61470	08/10/2003	201	171.129597015
61476	08/10/2003	219	159.070630137
61482	08/10/2003	175	164.45212
61488	08/10/2003	174	173.059247126
...			
...			
60888	08/12/2003	227	167.847951542
60894	08/12/2003	208	164.324375
60900	08/12/2003	202	172.871608911
60906	08/12/2003	225	170.849111111
60912	08/12/2003	201	176.565079602
60918	08/12/2003	211	160.253549763
60924	08/12/2003	198	183.566585859
60930	08/12/2003	239	179.631732218
60936	08/12/2003	222	179.888725225
60942	08/12/2003	229	167.542065502
60948	08/12/2003	180	171.243477778
60954	08/12/2003	210	185.098704762
60960	08/12/2003	221	181.685891403
60966	08/12/2003	215	186.954274419
60972	08/12/2003	218	185.699004587
60978	08/12/2003	229	176.988882096
60984	08/12/2003	264	163.267109848
60990	08/12/2003	216	178.108861111
60996	08/12/2003	227	183.128718062
61002	08/12/2003	203	181.477093596
61008	08/12/2003	232	176.856918103
61014	08/12/2003	216	182.383217593
61020	08/12/2003	208	172.684408654
61026	08/12/2003	205	174.99964878
61032	08/12/2003	224	181.347888393
61038	08/12/2003	238	171.833462185
61044	08/12/2003	210	193.40702381
61050	08/12/2003	196	169.058367347
61056	08/12/2003	205	188.203278049
61062	08/12/2003	206	175.217131068
61068	08/12/2003	222	166.161481982
61074	08/12/2003	196	167.21905102
61080	08/12/2003	217	169.222105991
61086	08/12/2003	226	179.661973451
61092	08/12/2003	198	164.571858586
61098	08/12/2003	201	177.023129353
61104	08/12/2003	219	176.921538813
61110	08/12/2003	221	170.655909502
61116	08/12/2003	207	186.579217391
61122	08/12/2003	205	162.04755122
61128	08/12/2003	195	178.919292308
61134	08/12/2003	219	179.041305936
61140	08/12/2003	201	189.613288557
61146	08/12/2003	207	162.462342995
61152	08/12/2003	208	161.918548077
61158	08/12/2003	247	165.20059919

61164	08/12/2003	195	172.864466667
61170	08/12/2003	186	178.453731183
61176	08/12/2003	197	169.326126904
61182	08/12/2003	212	180.578919811
61188	08/12/2003	204	166.141882353

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B PYTHON SCRIPTS

The following is the program code of the python script, filter.py, mentioned in Data Collection section of this thesis.

```
#!/usr/bin/python

import sys, re, getopt

def printcolumns(columns, separator = "\t"):
    for i in range(len(columns)):
        sys.stdout.write(columns[i])
        if i < len(columns) - 1:
            sys.stdout.write(separator)
        else:
            sys.stdout.write("\n")

def nextcolumns(inputfile, signatures, columns):
    sigmatch = 0

    line = inputfile.readline()
    while line:
        match = re.match(signatures[sigmatch], line)
        if match:
            columns[sigmatch] = match.group(1)
            sigmatch += 1

        match = re.match(signatures[0], line)
        if match:
            columns[0] = match.group(1)
            sigmatch = 1

        if sigmatch >= len(columns):
            return 1

        line = inputfile.readline()

    return 0

def usage():
    print sys.argv[0] + " [--rtt|--um] <filename>"
    print
    print "  Use --rtt for Rtt style files, and --um for UpdateMessages
style files"
    print
    print "  Produces columns separated by tabs."
    return

if __name__ == "__main__":
    if len(sys.argv) < 2:
        usage()
        sys.exit()
```

```
long_options = ['rtt', 'um']

arguments = getopt.getopt(sys.argv[1:], None, long_options)

input_file = open(arguments[1][0], 'r')

signatures = None

if len(arguments[0]) == 0:
    usage()
    sys.exit()

columns = None

if arguments[0][0][0] == '--rtt':
    # Initially empty list of columns
    columns = [None, None]
    signatures = ["\s*(creation:.*)$",
                  "\s*(Rtt:\s[^0].*)$"]
elif arguments[0][0][0] == '--um':
    # Initially empty list of columns
    columns = [None, None]
    signatures = ["\s*(TIME:.*)$",
                  "\s*((?:ANNOUNCE.*)|(?:WITHDRAW.*))$"]

while nextcolumns(input_file, signatures, columns):
    printcolumns(columns)
```

The following is the program code of the python script, timeslice.py, mentioned in Data Collection section of this thesis.

```
#!/usr/bin/python

import sys, re, getopt

RECORD_SEPARATOR = "\s*"

def get(value, colspec, separator, line):
    columns = re.split(separator, line)

    valueindex = colspec[value]
    if valueindex == None:
        return None
    return columns[valueindex]

def seconds(timeenc):
    components = re.split(":", timeenc)

    return int(components[2]) + int(components[1])*60 +
int(components[0])*60*60

def usage():
    print sys.argv[0] + " [--rtt|--um] [--ave|--med] [--slice=<n>]
<filename>"
    print
    print "  Use --rtt for Rtt style files, and --um for UpdateMessages
style files"
    print "  (default is Rtt style)"
    print
    print "  Use --ave to print out data means, and --med for medians"
    print "  (default is medians)"
    print
    print "  Specify a time slice over which to operate with --slice"
    print "  (in seconds, default is 6)"
    print
    print "  Produces a column with the total number of values within"
    print "  the six second range, and another with the average value"
    print "  within that range."
    print
    print "  This version divides the time literally every 6 seconds."
    return

if __name__ == "__main__":
    if len(sys.argv) < 2:
        usage()
        sys.exit()

    long_options = ['rtt', 'um', 'ave', 'med', 'slice=']

    arguments = getopt.getopt(sys.argv[1:], None, long_options)

    input_file = open(arguments[1][0], 'r')
```

```
    mode = 'median'
    colspec = {'date': 1, 'time': 2, 'value': 5}
    timeslice = 6
    for argument in arguments[0]:
        if argument[0] == '--rtt':
            colspec = {'date': 1, 'time': 2, 'value': 5}
        if argument[0] == '--um':
            colspec = {'date': 1, 'time': 2, 'value': None}
        if argument[0] == '--ave':
            mode = 'mean'
        if argument[0] == '--med':
            mode = 'median'
        if argument[0] == '--slice':
            timeslice = int(argument[1])

    line = input_file.readline()
    date0 = get('date', colspec, RECORD_SEPARATOR, line)
    time0 = get('time', colspec, RECORD_SEPARATOR, line)
    seconds0 = seconds(time0)
    records = 0
    recordlist = []
    total = 0.0
    nototal = 0
    value = 0.0
    output = None

    while line:
        if get('value', colspec, RECORD_SEPARATOR, line):
            value = float(get('value', colspec, RECORD_SEPARATOR,
line))
        else:
            nototal = 1
        date1 = get('date', colspec, RECORD_SEPARATOR, line)
        time1 = get('time', colspec, RECORD_SEPARATOR, line)
        seconds1 = seconds(time1)
        output = str(seconds0) + "\t\t" + str(date0) + "\t\t" +
str(records)
        if seconds1 > seconds0 + timeslice - 1 or date1 != date0:
            if not nototal:
                if mode == 'mean':
                    output += "\t\t" + str(total/records)
                elif mode == 'median':
                    recordlist.sort()
                    output += "\t\t" +
str(recordlist[len(recordlist)/2])
            print output

            emptyslices = 1
            nextdatatime = seconds0 + ((seconds1 - seconds0) /
timeslice) * timeslice
            while seconds0 + timeslice * emptyslices < nextdatatime:
                  print str(seconds0 + timeslice * emptyslices) +
"\t\t" + str(date0) + "\t\t0\t\tNo data"
                emptyslices += 1
            seconds0 = nextdatatime
```

```
            records = 1
            recordlist = [value]
            total = value
            nototal = 0
            if date1 != date0:
                date0 = date1
                seconds0 = seconds1
        else:
            recordlist.append(value)
            total += value
            records += 1

        line = input_file.readline()
    else:
        output = str(seconds0) + "\t\t" + str(date0) + "\t\t" +
str(records)
        if not nototal:
            if mode == 'mean':
                output += "\t\t" + str(total/records)
            elif mode == 'median':
                recordlist.sort()
                output += "\t\t" + str(recordlist[len(recordlist)/2])
        print output
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C MATLAB CORRELATION RESULTS

The following is the Matlab outputs of the max correlation coefficient and its lag and min correlation coefficient and its lag for various EMA window sizes of all three periods: East Coast Blackout, Blaster Worm and February 11-13. We show the EMA in the thesis. However, the following information will also show the results from Simple Moving Average for some comparison.

Feb_12and13_Movavg_Correlation_Avg	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff = 0.1134 k_value = -885	max_value_coeff = 0.1134 k_value = -885
	min_value_coeff = -0.0507 k_value = 22532	min_value_coeff = -0.0507 k_value = 22532
2) movingWindow = 2		
	max_value_coeff = 0.1514 k_value = -885	max_value_coeff = 0.1521 k_value = -885
	min_value_coeff = -0.0673 k_value = 22532	min_value_coeff = -0.0669 k_value = 22532
3) movingWindow = 8		
	max_value_coeff = 0.2216 k_value = -886	max_value_coeff = 0.2273 k_value = -885
	min_value_coeff = -0.0945 k_value = 22533	min_value_coeff = -0.0962 k_value = 22532
4) movingWindow = 32		
	max_value_coeff = 0.2941 k_value = -887	max_value_coeff = 0.3059 k_value = -888
	min_value_coeff = -0.1220 k_value = 26085	min_value_coeff = -0.1264 k_value = 26085
5) movingWindow = 128		
	max_value_coeff = 0.4064 k_value = -880	max_value_coeff = 0.4103 k_value = -888
	min_value_coeff = -0.1738 k_value = 26028	min_value_coeff = -0.1775 k_value = 26040

6) movingWindow = 512		
	max_value_coeff = 0.5101 k_value = -887	max_value_coeff = 0.5032 k_value = 1422
	min_value_coeff = -0.2434 k_value = 26025	min_value_coeff = -0.2542 k_value = 26047
7) movingWindow = 2048		
	max_value_coeff = 0.7240 k_value = 1855	max_value_coeff = 0.6677 k_value = 1411
	min_value_coeff = -0.3496 k_value = 26210	min_value_coeff = -0.3496 k_value = 25921
8) movingWindow = 8192		
	max_value_coeff = 0.9069 k_value = 625	max_value_coeff = 0.6725 k_value = 125
	min_value_coeff = -0.4962 k_value = 24763	min_value_coeff = -0.6588 k_value = -7137
Feb_12and13_Movavg_Correlation_Median	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff =0.0825 k_value = -884	max_value_coeff = 0.0825 k_value = -884
	min_value_coeff = -0.0410 k_value = 25941	min_value_coeff = -0.0410 k_value = 25941
2) movingWindow = 2		
	max_value_coeff = 0.1143 k_value = -885	max_value_coeff = 0.1142 k_value = -885
	min_value_coeff = -0.0553 k_value = 25941	min_value_coeff = -0.0551 k_value = 25941
3) movingWindow = 8		
	max_value_coeff = 0.1733 k_value = -886	max_value_coeff = 0.1767 k_value = -885
	min_value_coeff = -0.0833 k_value = 30991	min_value_coeff = -0.0844 k_value = 30991
4) movingWindow = 32		
	max_value_coeff = 0.2314 k_value = -882	max_value_coeff = 0.2397 k_value = -885

	min_value_coeff = -0.1116 k_value = 30790	min_value_coeff = -0.1148 k_value = 30786
5) movingWindow = 128		
	max_value_coeff = 0.3232 k_value = -865	max_value_coeff = 0.3237 k_value = -879
	min_value_coeff = -0.1558 k_value = 30794	min_value_coeff = -0.1527 k_value = 31023
6) movingWindow = 512		
	max_value_coeff = 0.4202 k_value = 3369	max_value_coeff = 0.4168 k_value = 1665
	min_value_coeff = -0.2202 k_value = 30995	min_value_coeff = -0.2052 k_value = 26020
7) movingWindow = 2048		
	max_value_coeff = 0.6807 k_value = 2265	max_value_coeff = 0.5444 k_value = 1763
	min_value_coeff = -0.2896 k_value = 26335	min_value_coeff = -0.4017 k_value = -5262
8) movingWindow = 8192		
	max_value_coeff = 0.8837 k_value = 1579	max_value_coeff = 0.4798 k_value = 55
	min_value_coeff = -0.4468 k_value = 25392	min_value_coeff = -0.6391 k_value = -7021
Blaster_Movavg_Correlation_Median	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff = 0.0626 k_value = 5893	max_value_coeff = 0.0626 k_value = 5893
	min_value_coeff = -0.0594 k_value = -2285	min_value_coeff = -0.0594 k_value = -2285
2) movingWindow = 2		
	max_value_coeff = 0.0794 k_value = 5893	max_value_coeff = 0.0799 k_value = 5893
	min_value_coeff = -0.0772 k_value = -1901	min_value_coeff = -0.0774 k_value = -1901

3) movingWindow = 8		
	max_value_coeff = 0.1139 k_value = 5894	max_value_coeff = 0.1165 k_value = 5894
	min_value_coeff = -0.1113 k_value = -1900	min_value_coeff = -0.1141 k_value = -1901
4) movingWindow = 32		
	max_value_coeff = 0.1513 k_value = 5904	max_value_coeff = 0.1567 k_value = 5920
	min_value_coeff = -0.1474 k_value = -1899	min_value_coeff = -0.1484 k_value = -1899
5) movingWindow = 128		
	max_value_coeff = 0.2208 k_value = 5905	max_value_coeff = 0.2174 k_value = 5903
	min_value_coeff = -0.1953 k_value = -1898	min_value_coeff = -0.2156 k_value = -5653
6) movingWindow = 512		
	max_value_coeff = 0.3512 k_value = 5833	max_value_coeff = 0.2988 k_value = 5860
	min_value_coeff = -0.3842 k_value = -1889	min_value_coeff = -0.4093 k_value = -5656
7) movingWindow = 2048		
	max_value_coeff = 0.5906 k_value = 5259	max_value_coeff = 0.2928 k_value = 5307
	min_value_coeff = -0.6558 k_value = -1925	min_value_coeff = -0.6450 k_value = -5654
8) movingWindow = 8192		
	max_value_coeff = 0.7490 k_value = 4266	max_value_coeff = 0.5591 k_value = 0
	min_value_coeff = -0.7817 k_value = -2082	min_value_coeff = -0.5443 k_value = -5655
Blaster_Movavg_Correlation_avg	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff = 0.0510 k_value = 3739	max_value_coeff = 0.051 k_value = 3739

	min_value_coeff = -0.0569 k_value = -1880	min_value_coeff = -0.0569 k_value = -1880
2) movingWindow = 2		
	max_value_coeff = 0.0663 k_value = 3739	max_value_coeff = 0.0669 k_value = 3739
	min_value_coeff = -0.0707 k_value = -1880	min_value_coeff = -0.0706 k_value = -1880
3) movingWindow = 8		
	max_value_coeff = 0.0964 k_value = 3740	max_value_coeff = 0.0986 k_value = 3739
	min_value_coeff = -0.1014 k_value = -1897	min_value_coeff = -0.1030 k_value = -1895
4) movingWindow = 32		
	max_value_coeff = 0.1277 k_value = 3739	max_value_coeff = 0.1277 k_value = 3739
	min_value_coeff = -0.1365 k_value = -1890	min_value_coeff = -0.1377 k_value = -1893
5) movingWindow = 128		
	max_value_coeff = 0.1862 k_value = 4309	max_value_coeff = 0.1892 k_value = 4263
	min_value_coeff = -0.1913 k_value = -2085	min_value_coeff = -0.2035 k_value = -5649
6) movingWindow = 512		
	max_value_coeff = 0.3181 k_value = 4173	max_value_coeff = 0.2989 k_value = 4200
	min_value_coeff = -0.3652 k_value = -2091	min_value_coeff = -0.3731 k_value = -5654
7) movingWindow = 2048		
	max_value_coeff = 0.5840 k_value = 4555	max_value_coeff = 0.3760 k_value = 4201
	min_value_coeff = -0.6019 k_value = -2199	min_value_coeff = -0.6325 k_value = -5651
8) movingWindow = 8192		
	max_value_coeff = 0.7888 k_value = 3586	max_value_coeff = 0.5909 k_value = 0

	min_value_coeff = -0.6718 k_value = -2859	min_value_coeff = -0.5546 k_value = -5653
ECB_Movavg_Correlation_Median	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff = 0.0461 k_value = 20267	max_value_coeff = 0.0461 k_value = 20267
	min_value_coeff = -0.0324 k_value = -12165	min_value_coeff = -0.0324 k_value = -12165
2) movingWindow = 2		
	max_value_coeff = 0.0617 k_value = 20267	max_value_coeff = 0.0627 k_value = 20266
	min_value_coeff = -0.0402 k_value = -14344	min_value_coeff = -0.0405 k_value = -13043
3) movingWindow = 8		
	max_value_coeff = 0.0936 k_value = 20265	max_value_coeff = 0.0970 k_value = 20259
	min_value_coeff = -0.0588 k_value = -13779	min_value_coeff = -0.0604 k_value = -13787
4) movingWindow = 32		
	max_value_coeff = 0.1397 k_value = 20264	max_value_coeff = 0.1443 k_value = 20235
	min_value_coeff = -0.0828 k_value = -13036	min_value_coeff = -0.0863 k_value = -13072
5) movingWindow = 128		
	max_value_coeff = 0.2139 k_value = 20289	max_value_coeff = 0.2165 k_value = 20154
	min_value_coeff = -0.1329 k_value = -13041	min_value_coeff = -0.1365 k_value = -13169
6) movingWindow = 512		
	max_value_coeff = 0.3260 k_value = 20457	max_value_coeff = 0.3263 k_value = 19847
	min_value_coeff = -0.1984 k_value = -14176	min_value_coeff = -0.2080 k_value = -13569

7) movingWindow = 2048		
	max_value_coeff = 0.4607 k_value = 20648	max_value_coeff = 0.4307 k_value = 18512
	min_value_coeff = -0.3288 k_value = -13088	min_value_coeff = -0.3684 k_value = -14758
8) movingWindow = 8192		
	max_value_coeff = 0.4718 k_value = 7149	max_value_coeff = 0.6329 k_value = -397
	min_value_coeff = -0.6064 k_value = -10490	min_value_coeff = -0.6516 k_value = -14815
ECB_Movavg_Correlation_avg	**Simple**	**Exponential**
1) movingWindow = 1		
	max_value_coeff = 0.0592 k_value = 7377	max_value_coeff = 0.0592 k_value = 7377
	min_value_coeff = -0.0355 k_value = -14507	min_value_coeff = -0.0355 k_value = -14507
2) movingWindow = 2		
	max_value_coeff = 0.0717 k_value = 7387	max_value_coeff = 0.0729 k_value = 7387
	min_value_coeff = -0.0420 k_value = 14871	min_value_coeff = -0.0429 k_value = 14870
3) movingWindow = 8		
	max_value_coeff = 0.0994 k_value = 7387	max_value_coeff = 0.1034 k_value = 7380
	min_value_coeff = -0.0576 k_value = 14886	min_value_coeff = -0.0596 k_value = 14879
4) movingWindow = 32		
	max_value_coeff = 0.1434 k_value = 7385	max_value_coeff = 0.1486 k_value = 7352
	min_value_coeff = -0.0824 k_value = 14885	min_value_coeff = -0.0851 k_value = 15068
5) movingWindow = 128		
	max_value_coeff = 0.2285 k_value = 20726	max_value_coeff = 0.2326 k_value = 20562

	min_value_coeff = -0.1334 k_value = 15109	min_value_coeff = -0.1375 k_value = 14975
6) movingWindow = 512		
	max_value_coeff = 0.3564 k_value = 20571	max_value_coeff = 0.3538 k_value = 20039
	min_value_coeff = -0.2067 k_value = 15217	min_value_coeff = -0.2025 k_value = 14804
7) movingWindow = 2048		
	max_value_coeff = 0.4845 k_value = 20643	max_value_coeff = 0.4540 k_value = 18569
	min_value_coeff = -0.2907 k_value = -13590	min_value_coeff = -0.3257 k_value = -14538
8) movingWindow = 8192		
	max_value_coeff = 0.4382 k_value = 7202	max_value_coeff = 0.6377 k_value = -335
	min_value_coeff = -0.5705 k_value = -10252	min_value_coeff = -0.6312 k_value = -14421

The following is the graphs of EMA RTT averages and EMA total updates of the three periods of interest all with moving window size of 512 samples.

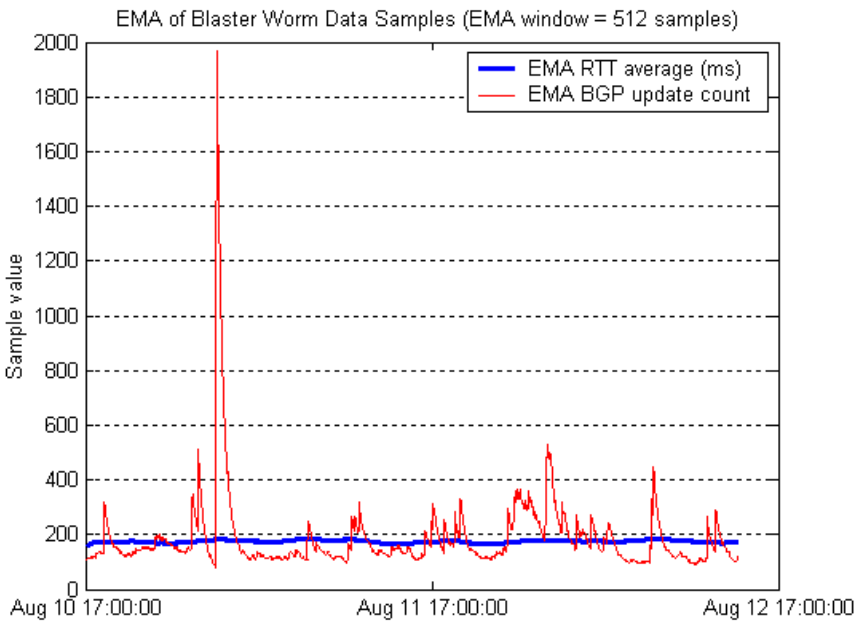

Figure 7. Blaster Worm Data

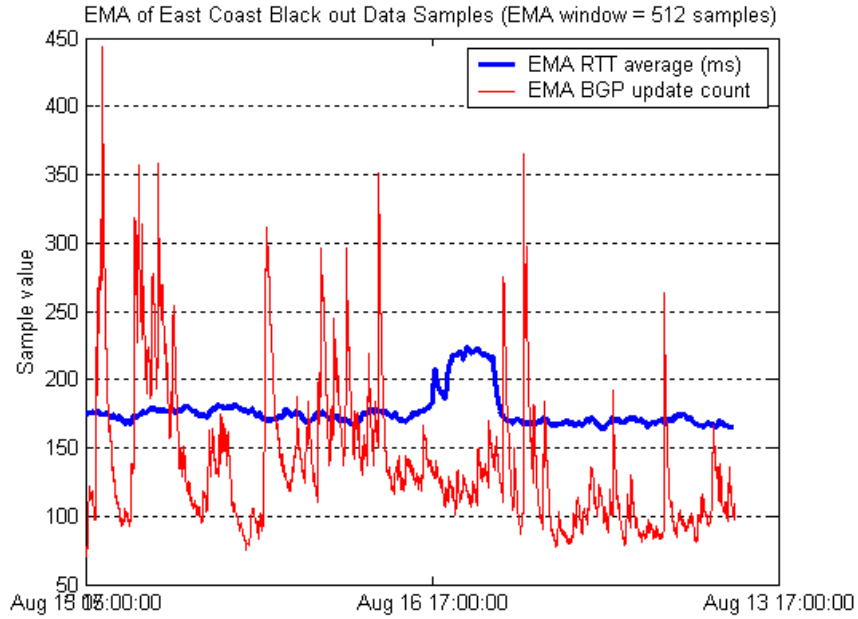

Figure 8. East Coast Blackout Data

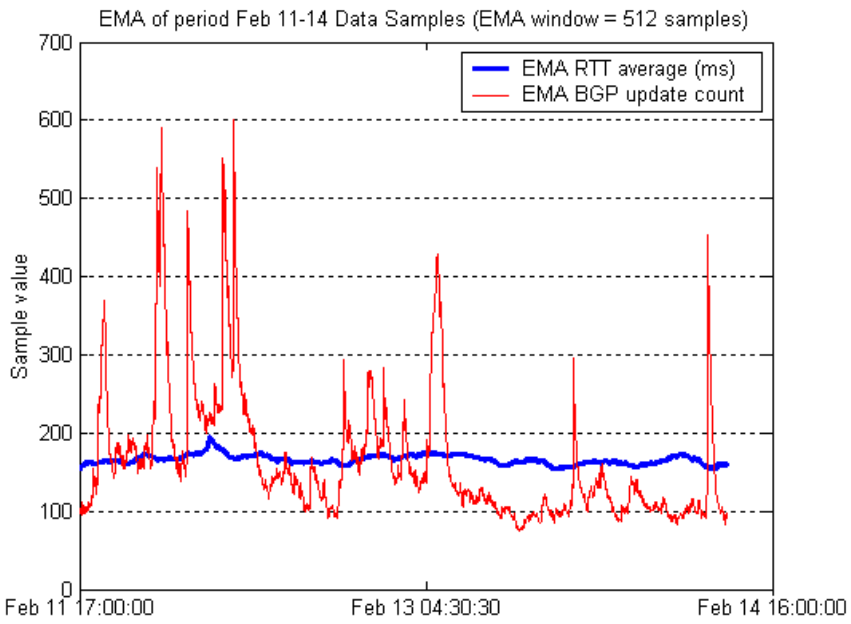

Figure 9. Feb 11-13

The following are the graphs of correlation of the total updates with RTT averages for the three time periods: Blaster Worm, East Coast Blackout and February 11-13.

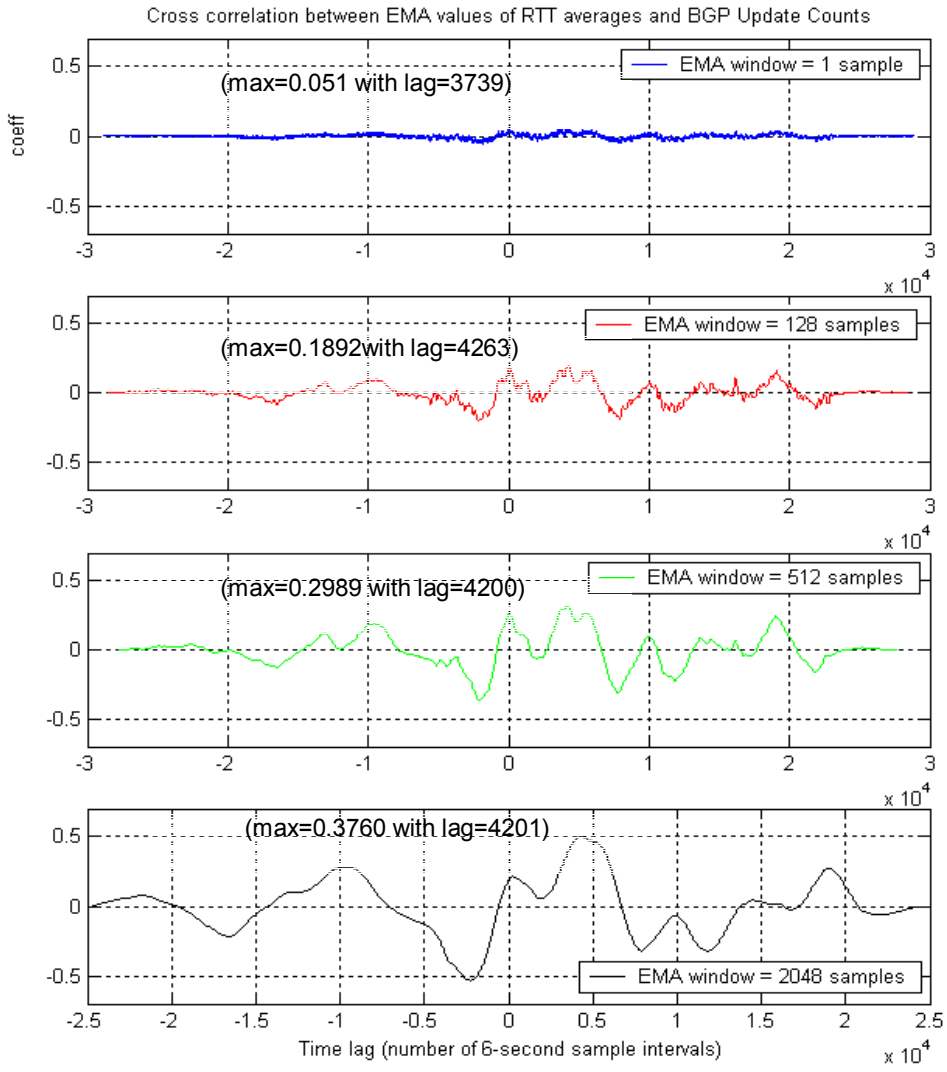

Figure 10. Correlation of RTT and total updates for the Blaster Worm period

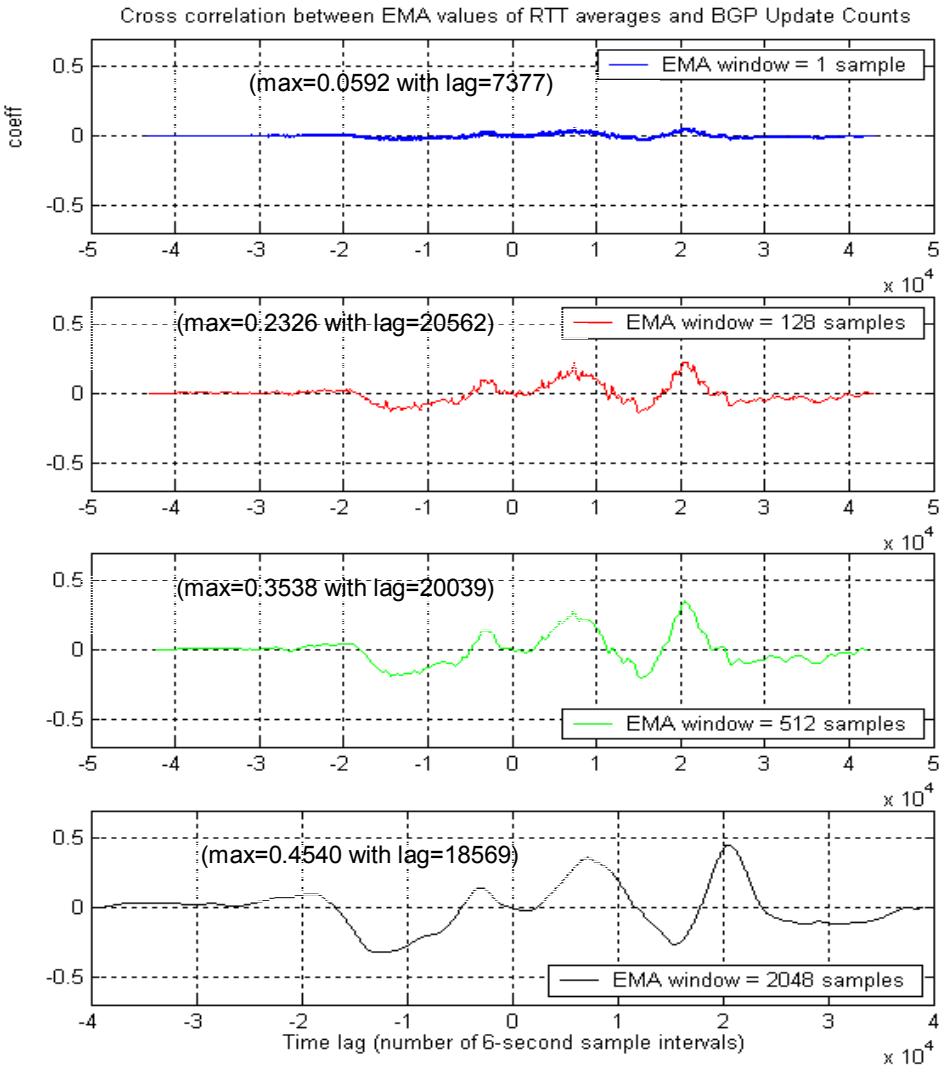

Figure 11. Correlation of RTT and total updates for the East Coast Blackout period

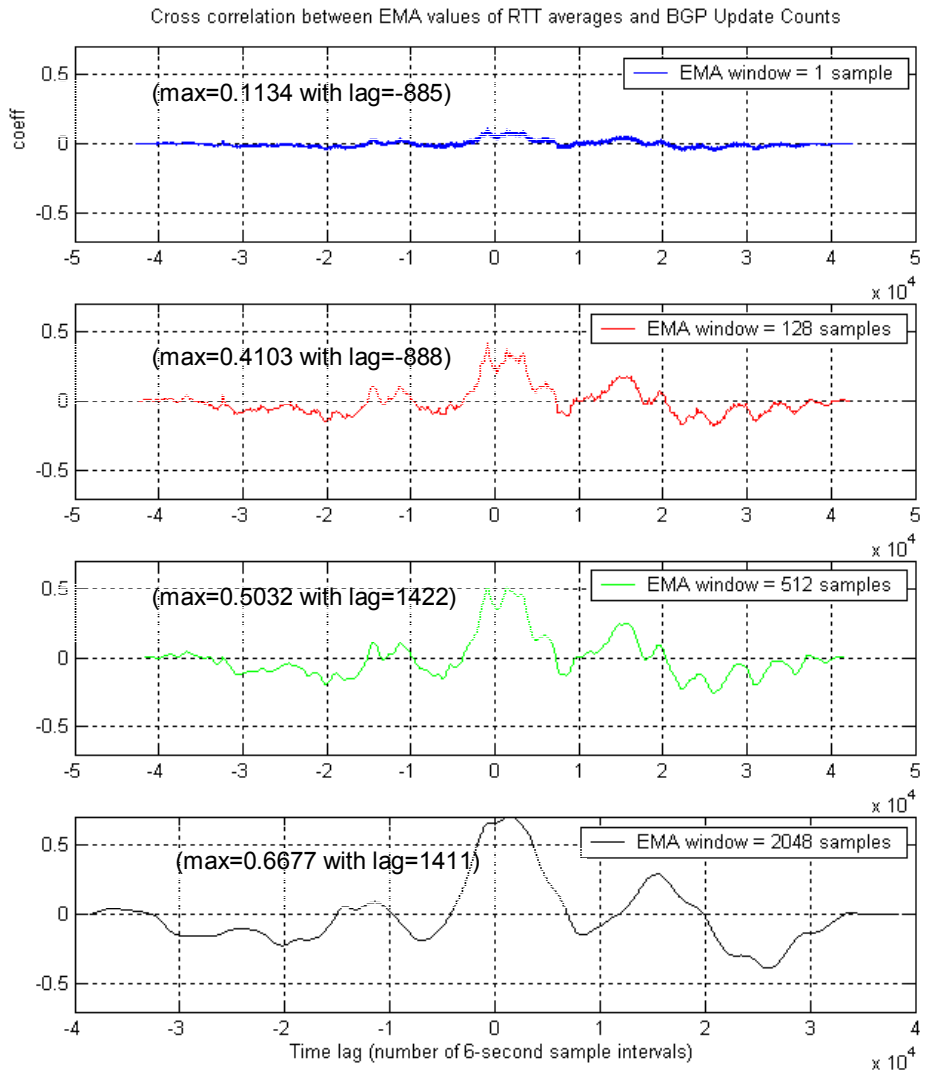

Figure 12. Correlation of RTT and total updates for the Feb 11-13 period

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D MATLAB PROGRAM CODE FOR CORRELATION AND GRAPH COMPUTATIONS

The following is the Matlab program code that was used to calculate the EMA and the correlation of the RTT and total updates. All graphs shown in this thesis were generated by this same program. This program calculates the Simple Moving Average, Linear Moving Average and Exponential Moving average of the RTT and total updates.

```
load RTT.txt; % Should comment this line out when we run the program the second time because RTT.txt was

                        %already loaded the first time the program was run.
load Updates.txt; % Should comment this line out when we run the program the second time because Updates.txt
                          %was already loaded the first time the program was run.
movingWindow=8192  %number of samples for computing the average
dummyWindow=1; %we use lagging average; so this window size doesn't matter
[short,movavg_0_RTT]=movavg(RTT, dummyWindow, movingWindow, 0);%option 0 in moving average would
                                                    %calculate Simple Moving Average
[short,movavg_0_Updates]=movavg(Updates, dummyWindow, movingWindow, 0);

movavg_0_RTT=movavg_0_RTT(movingWindow:end);
movavg_0_Updates=movavg_0_Updates(movingWindow:end);
[k,lags]=xcov(movavg_0_RTT,movavg_0_Updates,'coeff'); %-- this will give us 2 result vectors: one is the k of
             %-- correlation values and the other is of lag values.--%
             %-- 'biased' option is what we want because it divides the sum by N when it calculates the Rxy value  --%
 %-- this way the endpoints don't have to suffer from large variance.  We use exactly the same formula in our thesis--%
            %dlmwrite('result.txt',k,' ');%-- writing the "k" vector into a file called "result.txt" --%
plot(movavg_0_RTT)
hold on
plot(movavg_0_Updates,'r')
figure
plot(lags,k)    %-- plotting lags as X axis and k as Y axis...this way we can see where the highest correlation value
                %happens at what lag value --%
grid on %--turn the grid on so we can see the values from the graph more easily --%
max_value_coeff=max(k) %--this gives us the max Rxy value --%
k_value=lags(find(k==max_value_coeff)) %--this tells us at what lag value does the max Rxy happens --%
min_value_coeff = min(k)
k_value=lags(find(k==min_value_coeff))
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[short,movavg_1_RTT]=movavg(RTT, dummyWindow, movingWindow, 1);
%option 1 in moving average would calculate Linear Moving Average
[short,movavg_1_Updates]=movavg(Updates, dummyWindow, movingWindow, 1);
movavg_0_RTT=movavg_0_RTT(movingWindow:end);
movavg_0_Updates=movavg_0_Updates(movingWindow:end);
[k1,lags1]=xcov(movavg_1_RTT,movavg_1_Updates,'coeff');
%dlmwrite('result.txt',k,' ');
figure
plot(movavg_1_RTT)
hold on
plot(movavg_1_Updates,'r')
figure
plot(lags1,k1)
grid on
max_value_coeff=max(k1)
k_value=lags(find(k1==max_value_coeff))
min_value_coeff = min(k1)
k_value=lags(find(k1==min_value_coeff))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

[short,movavg_e_RTT]=movavg(RTT, dummyWindow, movingWindow, 'e'); %option 'e' in moving average would calculate Exponential Moving Average
[short,movavg_e_Updates]=movavg(Updates, dummyWindow, movingWindow, 'e');
movavg_0_RTT=movavg_0_RTT(movingWindow:end);
movavg_0_Updates=movavg_0_Updates(movingWindow:end);
[k2,lags2]=xcov(movavg_e_RTT,movavg_e_Updates,'coeff');
%dlmwrite('result.txt',k,' ');
figure
plot(movavg_e_RTT)
hold on
plot(movavg_e_Updates,'r')
figure
plot(lags2,k2)
grid on
max_value_coeff=max(k2)
k_value=lags(find(k2==max_value_coeff))
min_value_coeff = min(k2)
k_value=lags(find(k2==min_value_coeff))
```

APPENDIX E PATCH FILE FOR ARTS++ RELEASE 1-1-A9

The following is the patch file (diff file) for the arts++ release 1-1-A9 mentioned in Data Collection section of the thesis.

```
Only in arts++-1-1-a9.modified.backup: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsasagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsases: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsdump: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsintfmagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsintfms: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsnetagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsnets: Makefile
diff -u -r arts++-1-1-a9/apps/artsnets/artsnets.cc arts++-1-1-a9.modified.backup/apps/artsnets/artsnets.cc
--- arts++-1-1-a9/apps/artsnets/artsnets.cc             2002-11-15 16:46:38.000000000 -0800
+++ arts++-1-1-a9.modified.backup/apps/artsnets/artsnets.cc             2004-01-11 19:29:20.000000000 -0800
@@ -160,7 +160,7 @@
     }
   }

-  if (hostAttribute) {
+  if (hostAttribute != arts.Attributes().end()) {
     inAddr.s_addr = hostAttribute->Host();
     cout << "router:  " << inet_ntoa(inAddr) << endl;
   }
Only in arts++-1-1-a9.modified.backup/apps/artsnexthopagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsnexthops: Makefile
diff -u -r arts++-1-1-a9/apps/artsnexthops/artsnexthops.cc arts++-1-1-a9.modified.backup/apps/artsnexthops/artsnexthops.cc
--- arts++-1-1-a9/apps/artsnexthops/artsnexthops.cc             2002-11-15 16:46:39.000000000 -0800
+++ arts++-1-1-a9.modified.backup/apps/artsnexthops/artsnexthops.cc 2004-01-11 19:29:20.000000000 -0800
@@ -116,7 +116,7 @@
```

```
    }
  }

-  if (hostAttribute) {
+  if (hostAttribute != arts.Attributes().end()) {
     inAddr.s_addr = hostAttribute->Host();
     cout << "router:  " << inet_ntoa(inAddr) << endl;
  }
@@ -135,7 +135,7 @@
   cout << endl;

   periodAttribute = arts.FindPeriodAttribute();
-  if (periodAttribute) {
+  if (periodAttribute != arts.Attributes().end()) {
     periodTimes[0] = periodAttribute->Period()[0];
     periodTimes[1] = periodAttribute->Period()[1];
     localTm = localtime(&(periodTimes[0]));
Only in arts++-1-1-a9.modified.backup/apps/artsportagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsportmagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsportms: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsports: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsprotoagg: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artsprotos: Makefile
diff -u -r arts++-1-1-a9/apps/artsprotos/artsprotos.cc arts++-1-1-a9.modified.backup/apps/artsprotos/artsprotos.cc
--- arts++-1-1-a9/apps/artsprotos/artsprotos.cc       2002-11-15 16:46:39.000000000 -0800
+++ arts++-1-1-a9.modified.backup/apps/artsprotos/artsprotos.cc         2004-01-11 19:29:19.000000000 -0800
@@ -118,7 +118,7 @@
     }
   }

-  if (hostAttribute) {
+  if (hostAttribute != arts.Attributes().end()) {
     inAddr.s_addr = hostAttribute->Host();
```

```
    cout << "router:  " << inet_ntoa(inAddr) << endl;
  }
Only in arts++-1-1-a9.modified.backup/apps/artstoc: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artstos: Makefile
diff -u -r arts++-1-1-a9/apps/artstos/artstos.cc arts++-1-1-a9.modified.backup/apps/artstos/artstos.cc
--- arts++-1-1-a9/apps/artstos/artstos.cc 2002-11-15 16:46:39.000000000 -0800
+++ arts++-1-1-a9.modified.backup/apps/artstos/artstos.cc    2004-01-11 19:29:20.000000000 -0800
@@ -108,7 +108,7 @@
    }
  }

- if (hostAttribute) {
+  if (hostAttribute != arts.Attributes().end()) {
    inAddr.s_addr = hostAttribute->Host();
    cout << "router:  " << inet_ntoa(inAddr) << endl;
  }
@@ -126,7 +126,7 @@
  }
  cout << endl;

- if (periodAttribute) {
+  if (periodAttribute != arts.Attributes().end()) {
    periodTimes[0] = periodAttribute->Period()[0];
    periodTimes[1] = periodAttribute->Period()[1];
    localTm = localtime(&(periodTimes[0]));
Only in arts++-1-1-a9.modified.backup/apps/artstrunc: .libs
Only in arts++-1-1-a9.modified.backup/apps/artstrunc: Makefile
Only in arts++-1-1-a9.modified.backup/apps/artstrunc: artstrunc
Only in arts++-1-1-a9.modified.backup/apps/artstrunc: artstrunc.o
Only in arts++-1-1-a9.modified.backup/bsd.ports: Makefile
diff -u -r arts++-1-1-a9/classes/include/ArtsAsMatrixData.hh arts++-1-1-
a9.modified.backup/classes/include/ArtsAsMatrixData.hh
--- arts++-1-1-a9/classes/include/ArtsAsMatrixData.hh 2002-11-15 16:46:40.000000000 -0800
```

```
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsAsMatrixData.hh	2004-01-11 19:29:25.000000000 -0800
@@ -61,6 +61,8 @@

 #include "ArtsAsMatrixEntry.hh"

+using namespace std;
+
 //--------------------------------------------------------------------------
 //  class ArtsAsMatrixData
 //--------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/ArtsAttribute.hh arts++-1-1-a9.modified.backup/classes/include/ArtsAttribute.hh
--- arts++-1-1-a9/classes/include/ArtsAttribute.hh	2002-11-15 16:46:40.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsAttribute.hh	2004-01-11 19:29:25.000000000 -0800
@@ -50,6 +50,8 @@
 #include "caida_t.h"
 }

+using namespace std;
+
 #include <string>
 #include <istream.h>

diff -u -r arts++-1-1-a9/classes/include/ArtsBgp4AsPathSegment.hh arts++-1-1-a9.modified.backup/classes/include/ArtsBgp4AsPathSegment.hh
--- arts++-1-1-a9/classes/include/ArtsBgp4AsPathSegment.hh	2002-11-15 16:46:40.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsBgp4AsPathSegment.hh 2004-01-11 19:29:25.000000000 -0800
@@ -49,6 +49,8 @@

 #include <vector>

+using namespace std;
+
 //--------------------------------------------------------------------------
```

```
 //  class ArtsBgp4AsPathSegment

 //---------------------------------------------------------------------------

diff -u -r arts++-1-1-a9/classes/include/ArtsCflowdCustomData.hh arts++-1-1-a9.modified.backup/classes/include/ArtsCflowdCustomData.hh

--- arts++-1-1-a9/classes/include/ArtsCflowdCustomData.hh  2002-11-15 16:46:40.000000000 -0800

+++ arts++-1-1-a9.modified.backup/classes/include/ArtsCflowdCustomData.hh  2004-01-11 19:29:25.000000000 -0800

@@ -21,6 +21,8 @@


 #include <map>


+using namespace std;

+

 //---------------------------------------------------------------------------

 //                  class ArtsCflowdCustomDataKey

 //---------------------------------------------------------------------------

diff -u -r arts++-1-1-a9/classes/include/ArtsInterfaceMatrixData.hh arts++-1-1-a9.modified.backup/classes/include/ArtsInterfaceMatrixData.hh

--- arts++-1-1-a9/classes/include/ArtsInterfaceMatrixData.hh  2002-11-15 16:46:40.000000000 -0800

+++ arts++-1-1-a9.modified.backup/classes/include/ArtsInterfaceMatrixData.hh  2004-01-11 19:29:25.000000000 -0800

@@ -58,6 +58,8 @@


 #include "ArtsInterfaceMatrixEntry.hh"


+using namespace std;

+

 //---------------------------------------------------------------------------

 //  class ArtsInterfaceMatrixData

 //---------------------------------------------------------------------------

diff -u -r arts++-1-1-a9/classes/include/ArtsIpPathEntry.hh arts++-1-1-a9.modified.backup/classes/include/ArtsIpPathEntry.hh

--- arts++-1-1-a9/classes/include/ArtsIpPathEntry.hh  2003-05-28 22:42:57.000000000 -0700

+++ arts++-1-1-a9.modified.backup/classes/include/ArtsIpPathEntry.hh 2004-01-11 19:29:25.000000000 -0800

@@ -64,6 +64,8 @@

  #include <iomanip.h>
```

```
 #endif


+using namespace std;
+
 //---------------------------------------------------------------------------
 //     class ArtsIpPathEntry
 //---------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/ArtsNetMatrixEntry.hh arts++-1-1-
a9.modified.backup/classes/include/ArtsNetMatrixEntry.hh
--- arts++-1-1-a9/classes/include/ArtsNetMatrixEntry.hh 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsNetMatrixEntry.hh 2004-01-11 19:29:25.000000000 -0800
@@ -62,6 +62,8 @@
   #include <iomanip.h>
 #endif


+using namespace std;
+
 //---------------------------------------------------------------------------
 //  class ArtsNetMatrixEntry
 //---------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/ArtsPackageVersion.hh arts++-1-1-
a9.modified.backup/classes/include/ArtsPackageVersion.hh
--- arts++-1-1-a9/classes/include/ArtsPackageVersion.hh 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsPackageVersion.hh 2004-01-11 19:29:25.000000000 -0800
@@ -45,6 +45,8 @@


 #include <string>


+using namespace std;
+
 //---------------------------------------------------------------------------
 //  class ArtsPackageVersion
 //---------------------------------------------------------------------------
```

```
diff -u -r arts++-1-1-a9/classes/include/ArtsPortChoice.hh arts++-1-1-a9.modified.backup/classes/include/ArtsPortChoice.hh
--- arts++-1-1-a9/classes/include/ArtsPortChoice.hh 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsPortChoice.hh 2004-01-11 19:29:25.000000000 -0800
@@ -49,6 +49,8 @@

 #include <map>

+using namespace std;
+
 //----------------------------------------------------------------------------
 // class ArtsPortChoice
 //----------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/ArtsPortMatrixEntry.hh arts++-1-1-a9.modified.backup/classes/include/ArtsPortMatrixEntry.hh
--- arts++-1-1-a9/classes/include/ArtsPortMatrixEntry.hh 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsPortMatrixEntry.hh 2004-01-11 19:29:25.000000000 -0800
@@ -61,6 +61,8 @@
   #include <iomanip.h>
 #endif

+using namespace std;
+
 //----------------------------------------------------------------------------
 // class ArtsPortMatrixEntry
 //----------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/ArtsRttTimeSeriesTableData.hh arts++-1-1-a9.modified.backup/classes/include/ArtsRttTimeSeriesTableData.hh
--- arts++-1-1-a9/classes/include/ArtsRttTimeSeriesTableData.hh 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsRttTimeSeriesTableData.hh 2004-01-11 19:29:25.000000000 -0800
@@ -51,6 +51,8 @@

 #include <vector>
```

```
+using namespace std;
+
 //-------------------------------------------------------------------------
 //                  class ArtsRttTimeSeriesTableEntry
 //-------------------------------------------------------------------------
diff                -u                -r                arts++-1-1-a9/classes/include/ArtsSelectionSet.hh                arts++-1-1-
a9.modified.backup/classes/include/ArtsSelectionSet.hh
--- arts++-1-1-a9/classes/include/ArtsSelectionSet.hh           2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/ArtsSelectionSet.hh           2004-01-11 19:29:25.000000000 -0800
@@ -48,10 +48,13 @@
 }

 #include <vector>
+#include <iterator>
 #include <algorithm>

 #include "ArtsSelection.hh"

+using namespace std;
+
 //-------------------------------------------------------------------------
 //  template <class Type> class ArtsSelectionSet
 //-------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/include/Ipv4Network.hh arts++-1-1-a9.modified.backup/classes/include/Ipv4Network.hh
--- arts++-1-1-a9/classes/include/Ipv4Network.hh    2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/include/Ipv4Network.hh      2004-01-11 19:29:25.000000000 -0800
@@ -19,6 +19,8 @@
 #include <unistd.h>
 }

+using namespace std;
+
```

```
 #ifdef HAVE_IOSTREAM
  #include <iostream.h>
 #else
@@ -180,7 +182,7 @@
  //---------------------------------------------------------------------------
  istream & read(istream & is)
  {
-   is.read(&this->maskLen,sizeof(this->maskLen));
+   is.read(reinterpret_cast<char*>(&this->maskLen),sizeof(this->maskLen));
    uint8_t  octet1  = 0;
    uint8_t  octet2  = 0;
    uint8_t  octet3  = 0;
@@ -188,25 +190,25 @@


    switch (netSize) {
      case 1:
-       is.read(&octet1,sizeof(octet1));
+       is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
        this->net = htonl((ipv4addr_t)octet1 << 24);
        break;
      case 2:
-       is.read(&octet1,sizeof(octet1));
-       is.read(&octet2,sizeof(octet2));
+       is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+       is.read(reinterpret_cast<char*>(&octet2),sizeof(octet2));
        this->net =
          htonl(((ipv4addr_t)octet1 << 24) | ((ipv4addr_t)octet2 << 16));
        break;
      case 3:
-       is.read(&octet1,sizeof(octet1));
-       is.read(&octet2,sizeof(octet2));
-       is.read(&octet3,sizeof(octet3));
+       is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
```

```
+       is.read(reinterpret_cast<char*>(&octet2),sizeof(octet2));
+       is.read(reinterpret_cast<char*>(&octet3),sizeof(octet3));
        this->net = htonl(((ipv4addr_t)octet1 << 24) |
                          ((ipv4addr_t)octet2 << 16) |
                          ((ipv4addr_t)octet3 << 8));
        break;
      case 4:
-       is.read(&this->net,sizeof(this->net));
+       is.read(reinterpret_cast<char*>(&this->net),sizeof(this->net));
        break;
      default:
        break;
@@ -293,7 +295,7 @@
  ostream & write(ostream & os) const
  {
    //  first we write the netmask length
-   os.write(&this->maskLen,sizeof(this->maskLen));
+   os.write(reinterpret_cast<char*>(const_cast<uint8_t*>(&this->maskLen)),sizeof(this->maskLen));

    uint8_t  octet1 = 0;
    uint8_t  octet2 = 0;
@@ -305,26 +307,26 @@
    switch (netSize) {
      case 1:
        octet1 = ntohl(this->net) >> 24;
-       os.write(&octet1,sizeof(octet1));
+       os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
        break;
      case 2:
        netaddr = ntohl(this->net);
        octet1 = (ipv4addr_t)(netaddr >> 24) & 0xff;
        octet2 = (ipv4addr_t)(netaddr >> 16) & 0xff;
-       os.write(&octet1,sizeof(octet1));
```

```
-       os.write(&octet2,sizeof(octet2));
+       os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+       os.write(reinterpret_cast<char*>(&octet2),sizeof(octet2));
        break;
      case 3:
        netaddr = ntohl(this->net);
        octet1 = (ipv4addr_t)(netaddr >> 24) & 0xff;
        octet2 = (ipv4addr_t)(netaddr >> 16) & 0xff;
        octet3 = (ipv4addr_t)(netaddr >> 8) & 0xff;
-       os.write(&octet1,sizeof(octet1));
-       os.write(&octet2,sizeof(octet2));
-       os.write(&octet3,sizeof(octet3));
+       os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+       os.write(reinterpret_cast<char*>(&octet2),sizeof(octet2));
+       os.write(reinterpret_cast<char*>(&octet3),sizeof(octet3));
        break;
      case 4:
-       os.write(&this->net,sizeof(this->net));
+       os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&this->net)),sizeof(this->net));
        break;
      default:
        break;
Only in arts++-1-1-a9.modified.backup/classes/include: Ipv4PrefixPatricia.hh
Only in arts++-1-1-a9.modified.backup/classes/include: Makefile
Only in arts++-1-1-a9.modified.backup/classes/include: artslocal.h
Only in arts++-1-1-a9.modified.backup/classes/include: caida_t.h
diff -u -r arts++-1-1-a9/classes/src/ArtsAsMatrixAggregatorMap.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsAsMatrixAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsAsMatrixAggregatorMap.cc 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsAsMatrixAggregatorMap.cc    2004-01-11 19:29:24.000000000 -0800
@@ -45,6 +45,7 @@
 }
```

```
 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@

 #include "ArtsAsMatrixAggregatorMap.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsAsMatrixAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp $";

 //--------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsAsMatrixData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsAsMatrixData.cc
--- arts++-1-1-a9/classes/src/ArtsAsMatrixData.cc  2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsAsMatrixData.cc     2004-01-11 19:29:23.000000000 -0800
@@ -81,7 +81,7 @@
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-istream & ArtsAsMatrixData::read(istream& is, uint8_t version = 0)
+istream & ArtsAsMatrixData::read(istream& is, uint8_t version)
 {
   uint32_t          entryNum;
   ArtsAsMatrixEntry  asEntry;
@@ -106,7 +106,7 @@
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-int ArtsAsMatrixData::read(int fd, uint8_t version = 0)
+int ArtsAsMatrixData::read(int fd, uint8_t version)
```

```
{
  uint32_t          entryNum;
  ArtsAsMatrixEntry  asEntry;
@@ -161,7 +161,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-ostream & ArtsAsMatrixData::write(ostream& os, uint8_t version = 0)
+ostream & ArtsAsMatrixData::write(ostream& os, uint8_t version)
 {
   uint32_t       entryNum;

@@ -183,7 +183,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsAsMatrixData::write(int fd, uint8_t version = 0)
+int ArtsAsMatrixData::write(int fd, uint8_t version)
 {
   uint32_t       entryNum;
   int            rc;
@@ -235,7 +235,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-uint32_t ArtsAsMatrixData::Length(uint8_t version = 0) const
+uint32_t ArtsAsMatrixData::Length(uint8_t version) const
 {
   uint32_t  length = 0;

diff -u -r arts++-1-1-a9/classes/src/ArtsAsMatrixEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsAsMatrixEntry.cc
--- arts++-1-1-a9/classes/src/ArtsAsMatrixEntry.cc 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsAsMatrixEntry.cc    2004-01-11 19:29:21.000000000 -0800
```

```
@@ -49,6 +49,8 @@
 #include "ArtsAsMatrixEntry.hh"
 #include "ArtsPrimitive.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsAsMatrixEntry.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp $";

 //--------------------------------------------------------------------------
@@ -180,7 +182,7 @@
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-uint32_t ArtsAsMatrixEntry::Length(uint8_t version = 0) const
+uint32_t ArtsAsMatrixEntry::Length(uint8_t version) const
 {
   uint32_t   length;

@@ -193,18 +195,18 @@
 }

 //--------------------------------------------------------------------------
-//   istream& ArtsAsMatrixEntry::read(istream& is, uint8_t version = 0)
+//   istream& ArtsAsMatrixEntry::read(istream& is, uint8_t version)
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-istream& ArtsAsMatrixEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsAsMatrixEntry::read(istream& is, uint8_t version)
 {
   uint8_t        bytesize,
                  pktsize,
```

```
                srcsize,
                dstsize;

-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
   dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -224,7 +226,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsAsMatrixEntry::read(int fd, uint8_t version = 0)
+int ArtsAsMatrixEntry::read(int fd, uint8_t version)
 {
   uint8_t       bytesize,
                 pktsize,
@@ -277,14 +279,14 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-ostream & ArtsAsMatrixEntry::write(ostream & os, uint8_t version = 0) const
+ostream & ArtsAsMatrixEntry::write(ostream & os, uint8_t version) const
 {
   uint8_t       bytesize,
                 pktsize,
                 srcsize,
                 dstsize;

-  os.write(&this->_descriptor,sizeof(this->_descriptor));
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
```

```
  dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -304,7 +306,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsAsMatrixEntry::write(int fd, uint8_t version = 0) const
+int ArtsAsMatrixEntry::write(int fd, uint8_t version) const
 {
   uint8_t        bytesize,
                  pktsize,
diff -u -r arts++-1-1-a9/classes/src/ArtsAttribute.cc arts++-1-1-a9.modified.backup/classes/src/ArtsAttribute.cc
--- arts++-1-1-a9/classes/src/ArtsAttribute.cc          2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsAttribute.cc          2004-01-11 19:29:23.000000000 -0800
@@ -487,10 +487,10 @@

   idAndFormat = (this->_identifier << 8) | this->_format;
   uIntDatum = htonl(idAndFormat);
-  os.write(&uIntDatum,sizeof(uIntDatum));
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));

   uIntDatum = htonl(this->_length);
-  os.write(&uIntDatum,sizeof(uIntDatum));
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));

   switch (this->_identifier) {
     case artsC_ATTR_COMMENT:
@@ -499,16 +499,16 @@
       break;
     case artsC_ATTR_CREATION:
       uIntDatum = htonl(this->_value.creation);
-      os.write(&uIntDatum,sizeof(uIntDatum));
+      os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
       break;
```

```
    case artsC_ATTR_PERIOD:
      uIntDatum = htonl(this->_value.period[0]);
-     os.write(&uIntDatum,sizeof(uIntDatum));
+     os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
      uIntDatum = htonl(this->_value.period[1]);
-     os.write(&uIntDatum,sizeof(uIntDatum));
+     os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
      break;
    case artsC_ATTR_HOST:
-     os.write(&this->_value.host,sizeof(this->_value.host));
+     os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&this->_value.host)),sizeof(this->_value.host));
      break;
    case artsC_ATTR_IFDESCR:
      ptr = this->_value.ifDescr->c_str();
@@ -516,14 +516,14 @@
      break;
    case artsC_ATTR_IFINDEX:
      uShortDatum = htons(this->_value.ifIndex);
-     os.write(&uShortDatum,sizeof(uShortDatum));
+     os.write(reinterpret_cast<char*>(&uShortDatum),sizeof(uShortDatum));
      break;
    case artsC_ATTR_IFIPADDR:
-     os.write(&this->_value.ifIpAddr,sizeof(this->_value.ifIpAddr));
+     os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&this->_value.ifIpAddr)),sizeof(this->_value.ifIpAddr));
      break;
    case artsC_ATTR_HOSTPAIR:
-     os.write(&(this->_value.hostPair[0]),sizeof(ipv4addr_t));
-     os.write(&(this->_value.hostPair[1]),sizeof(ipv4addr_t));
+     os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&(this->_value.hostPair[0]))),sizeof(ipv4addr_t));
+     os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&(this->_value.hostPair[1]))),sizeof(ipv4addr_t));
      break;
    default:
      break;
```

```
@@ -675,12 +675,12 @@
      break;
  }

- is.read(&uIntDatum,sizeof(uIntDatum));
+ is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
  idAndFormat = ntohl(uIntDatum);
  this->_identifier = idAndFormat >> 8;
  this->_format = idAndFormat & 0xff;

- is.read(&uIntDatum,sizeof(uIntDatum));
+ is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
  this->_length = ntohl(uIntDatum);

  switch (this->_identifier) {
@@ -693,17 +693,17 @@
      free(ptr);
      break;
    case artsC_ATTR_CREATION:
-     is.read(&uIntDatum,sizeof(uIntDatum));
+     is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
      this->_value.creation = ntohl(uIntDatum);
      break;
    case artsC_ATTR_PERIOD:
-     is.read(&uIntDatum,sizeof(uIntDatum));
+     is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
      this->_value.period[0] = htonl(uIntDatum);
-     is.read(&uIntDatum,sizeof(uIntDatum));
+     is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
      this->_value.period[1] = htonl(uIntDatum);
      break;
    case artsC_ATTR_HOST:
-     is.read(&this->_value.host,sizeof(this->_value.host));
```

```
+      is.read(reinterpret_cast<char*>(&this->_value.host),sizeof(this->_value.host));
       break;
     case artsC_ATTR_IFDESCR:
       ptr = (char *)malloc(this->_length - (sizeof(uint32_t) * 2));
@@ -714,15 +714,15 @@
       free(ptr);
       break;
     case artsC_ATTR_IFINDEX:
-      is.read(&this->_value.ifIndex,sizeof(this->_value.ifIndex));
+      is.read(reinterpret_cast<char*>(&this->_value.ifIndex),sizeof(this->_value.ifIndex));
       this->_value.ifIndex = ntohs(this->_value.ifIndex);
       break;
     case artsC_ATTR_IFIPADDR:
-      is.read(&this->_value.ifIpAddr,sizeof(this->_value.ifIpAddr));
+      is.read(reinterpret_cast<char*>(&this->_value.ifIpAddr),sizeof(this->_value.ifIpAddr));
       break;
     case artsC_ATTR_HOSTPAIR:
-      is.read(&(this->_value.hostPair[0]),sizeof(ipv4addr_t));
-      is.read(&(this->_value.hostPair[1]),sizeof(ipv4addr_t));
+      is.read(reinterpret_cast<char*>(&(this->_value.hostPair[0])),sizeof(ipv4addr_t));
+      is.read(reinterpret_cast<char*>(&(this->_value.hostPair[1])),sizeof(ipv4addr_t));
       break;
     default:
       break;
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4AggregatorAttribute.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsBgp4AggregatorAttribute.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4AggregatorAttribute.cc2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4AggregatorAttribute.cc   2004-01-11 19:29:21.000000000 -0800
@@ -45,6 +45,8 @@
 #include "ArtsPrimitive.hh"
 #include "ArtsBgp4AggregatorAttribute.hh"

+using namespace std;
```

```
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsBgp4AggregatorAttribute.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp $";


 //---------------------------------------------------------------------------
@@ -154,19 +156,19 @@
 //
 //---------------------------------------------------------------------------
 istream & ArtsBgp4AggregatorAttribute::read(istream & is,
-                                            uint8_t version = 0)
+                                            uint8_t version)
 {
   g_ArtsLibInternal_Primitive.ReadUint16(is,this->_AS,sizeof(this->_AS));
-  is.read(&this->_ipAddr,sizeof(this->_ipAddr));
+  is.read(reinterpret_cast<char*>(&this->_ipAddr),sizeof(this->_ipAddr));
   return(is);
 }


 //---------------------------------------------------------------------------
-//    int ArtsBgp4AggregatorAttribute::read(int fd, uint8_t version = 0)
+//    int ArtsBgp4AggregatorAttribute::read(int fd, uint8_t version)
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AggregatorAttribute::read(int fd, uint8_t version = 0)
+int ArtsBgp4AggregatorAttribute::read(int fd, uint8_t version)
 {
   int  rc;
   int  bytesRead = 0;
@@ -190,19 +192,19 @@
 //
 //---------------------------------------------------------------------------
 ostream & ArtsBgp4AggregatorAttribute::write(ostream & os,
```

```
-                                       uint8_t version = 0) const
+                                        uint8_t version) const
 {
   g_ArtsLibInternal_Primitive.WriteUint16(os,this->_AS,sizeof(this->_AS));
-  os.write(&this->_ipAddr,sizeof(this->_ipAddr));
+  os.write(reinterpret_cast<const char*>(&this->_ipAddr),sizeof(this->_ipAddr));
   return(os);
 }

 //---------------------------------------------------------------------------
-// int ArtsBgp4AggregatorAttribute::write(int fd, uint8_t version = 0) const
+// int ArtsBgp4AggregatorAttribute::write(int fd, uint8_t version) const
 //...........................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AggregatorAttribute::write(int fd, uint8_t version = 0) const
+int ArtsBgp4AggregatorAttribute::write(int fd, uint8_t version) const
 {
   int  rc;
   int  bytesWritten = 0;
@@ -225,7 +227,7 @@
 //...........................................................................
 //
 //---------------------------------------------------------------------------
-uint32_t ArtsBgp4AggregatorAttribute::Length(uint8_t version = 0) const
+uint32_t ArtsBgp4AggregatorAttribute::Length(uint8_t version) const
 {
   return(sizeof(this->_AS) + sizeof(this->_ipAddr));
 }
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4AsPathAttribute.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4AsPathAttribute.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4AsPathAttribute.cc       2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4AsPathAttribute.cc         2004-01-11 19:29:23.000000000 -0800
```

```
@@ -117,7 +117,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream & ArtsBgp4AsPathAttribute::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4AsPathAttribute::read(istream & is, uint8_t version)
 {
   ArtsBgp4AsPathSegment  asPathSegment;
   uint8_t                numSegments;
@@ -125,7 +125,7 @@
   if (this->_segments.size() > 0)
     this->_segments.erase(this->_segments.begin(),this->_segments.end());

-  is.read(&numSegments,sizeof(numSegments));
+  is.read(reinterpret_cast<char*>(&numSegments),sizeof(numSegments));
   if (numSegments > 0) {
     this->_segments.reserve(numSegments);
     for (uint8_t segmentNum = 0; segmentNum < numSegments; segmentNum++) {
@@ -144,7 +144,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AsPathAttribute::read(int fd, uint8_t version = 0)
+int ArtsBgp4AsPathAttribute::read(int fd, uint8_t version)
 {
   int                  rc;
   int                  bytesRead = 0;
@@ -175,10 +175,10 @@
 //
 //---------------------------------------------------------------------------
 ostream & ArtsBgp4AsPathAttribute::write(ostream & os,
-                                         uint8_t version = 0) const
+                                         uint8_t version) const
```

```
 {
   uint8_t                numSegments = this->_segments.size();
-  os.write(&numSegments,sizeof(numSegments));
+  os.write(reinterpret_cast<char*>(&numSegments),sizeof(numSegments));
   for (uint8_t segmentNum = 0; segmentNum < numSegments; segmentNum++) {
     this->_segments[segmentNum].write(os,version);
   }
@@ -186,11 +186,11 @@
 }

 //---------------------------------------------------------------------------
-//  int ArtsBgp4AsPathAttribute::write(int fd, uint8_t version = 0) const
+//  int ArtsBgp4AsPathAttribute::write(int fd, uint8_t version) const
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AsPathAttribute::write(int fd, uint8_t version = 0) const
+int ArtsBgp4AsPathAttribute::write(int fd, uint8_t version) const
 {
   int        rc;
   int        bytesWritten = 0;
@@ -232,7 +232,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-uint32_t ArtsBgp4AsPathAttribute::Length(uint8_t version = 0) const
+uint32_t ArtsBgp4AsPathAttribute::Length(uint8_t version) const
 {
   uint32_t  length = sizeof(uint8_t);

diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4AsPathSegment.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4AsPathSegment.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4AsPathSegment.cc  2002-11-15 16:46:41.000000000 -0800
```

```
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4AsPathSegment.cc        2004-01-11 19:29:21.000000000 -0800
@@ -137,13 +137,13 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream & ArtsBgp4AsPathSegment::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4AsPathSegment::read(istream & is, uint8_t version)
 {
   uint8_t   numAses;
   uint16_t  as;

-  is.read(&this->_type,sizeof(this->_type));
-  is.read(&numAses,sizeof(numAses));
+  is.read(reinterpret_cast<char*>(&this->_type),sizeof(this->_type));
+  is.read(reinterpret_cast<char*>(&numAses),sizeof(numAses));
   if (numAses > 0) {
     this->_AS.reserve(numAses);
     for (int asNum = 0; asNum < numAses; asNum++) {
@@ -160,7 +160,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AsPathSegment::read(int fd, uint8_t version = 0)
+int ArtsBgp4AsPathSegment::read(int fd, uint8_t version)
 {
   int       rc;
   uint8_t   numAses;
@@ -196,12 +196,12 @@
 //
 //---------------------------------------------------------------------------
 ostream & ArtsBgp4AsPathSegment::write(ostream & os,
-                                       uint8_t version = 0) const
+                                       uint8_t version) const
```

```
 {
-  os.write(&this->_type,sizeof(this->_type));
+  os.write(reinterpret_cast<const char*>(&this->_type),sizeof(this->_type));

   uint8_t   numAses = this->_AS.size();
-  os.write(&numAses,sizeof(numAses));
+  os.write(reinterpret_cast<char*>(&numAses),sizeof(numAses));

   for (int asNum = 0; asNum < numAses; asNum++) {
     g_ArtsLibInternal_Primitive.WriteUint16(os,this->_AS[asNum],
@@ -215,7 +215,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4AsPathSegment::write(int fd, uint8_t version = 0) const
+int ArtsBgp4AsPathSegment::write(int fd, uint8_t version) const
 {
   int       rc;
   uint8_t   numAses;
@@ -246,7 +246,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-uint32_t ArtsBgp4AsPathSegment::Length(uint8_t version = 0) const
+uint32_t ArtsBgp4AsPathSegment::Length(uint8_t version) const
 {
   return(sizeof(this->_type) + sizeof(uint8_t) +
          (sizeof(uint16_t) * this->_AS.size()));
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4Attribute.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4Attribute.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4Attribute.cc  2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4Attribute.cc     2004-01-11 19:29:22.000000000 -0800
@@ -444,26 +444,26 @@
 //.....................................................................
```

```
 //
 //--------------------------------------------------------------------------
-istream & ArtsBgp4Attribute::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4Attribute::read(istream & is, uint8_t version)
 {
   uint8_t      numCommunities;
   uint32_t     community;


   this->DeleteValue();


-  is.read(&this->_flags,sizeof(this->_flags));
-  is.read(&this->_type,sizeof(this->_type));
+  is.read(reinterpret_cast<char*>(&this->_flags),sizeof(this->_flags));
+  is.read(reinterpret_cast<char*>(&this->_type),sizeof(this->_type));


   switch (this->_type) {
     case Bgp4_Attribute_Origin:
-      is.read(&this->_value._origin,sizeof(this->_value._origin));
+      is.read(reinterpret_cast<char*>(&this->_value._origin),sizeof(this->_value._origin));
       break;
     case Bgp4_Attribute_AsPath:
       this->_value._asPath = new ArtsBgp4AsPathAttribute;
       this->_value._asPath->read(is,version);
       break;
     case Bgp4_Attribute_NextHop:
-      is.read(&this->_value._nextHop,sizeof(this->_value._nextHop));
+      is.read(reinterpret_cast<char*>(&this->_value._nextHop),sizeof(this->_value._nextHop));
       break;
     case Bgp4_Attribute_MultiExitDisc:
       g_ArtsLibInternal_Primitive.ReadUint32(is,this->_value._MED,
@@ -480,7 +480,7 @@
       this->_value._aggregator->read(is,version);
       break;
```

```
    case Bgp4_Attribute_Community:
-     is.read(&numCommunities,sizeof(numCommunities));
+     is.read(reinterpret_cast<char*>(&numCommunities),sizeof(numCommunities));
      this->_value._community = new vector<uint32_t>;
      this->_value._community->reserve((int)numCommunities);
      for (int commNum = 0; commNum < numCommunities; commNum++) {
@@ -507,7 +507,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4Attribute::read(int fd, uint8_t version = 0)
+int ArtsBgp4Attribute::read(int fd, uint8_t version)
 {
   uint8_t      numCommunities;
   uint32_t     community;
@@ -622,22 +622,22 @@
 //
 //---------------------------------------------------------------------------
 ostream & ArtsBgp4Attribute::write(ostream & os,
-                                   uint8_t version = 0) const
+                                   uint8_t version) const
 {
   uint8_t  numCommunities;

-  os.write(&this->_flags,sizeof(this->_flags));
-  os.write(&this->_type,sizeof(this->_type));
+  os.write(reinterpret_cast<const char*>(&this->_flags),sizeof(this->_flags));
+  os.write(reinterpret_cast<const char*>(&this->_type),sizeof(this->_type));

    switch (this->_type) {
    case Bgp4_Attribute_Origin:
-     os.write(&this->_value._origin,sizeof(this->_value._origin));
+     os.write(reinterpret_cast<const char*>(&this->_value._origin),sizeof(this->_value._origin));
```

```
      break;
    case Bgp4_Attribute_AsPath:
      this->_value._asPath->write(os,version);
      break;
    case Bgp4_Attribute_NextHop:
-     os.write(&this->_value._nextHop,sizeof(this->_value._nextHop));
+     os.write(reinterpret_cast<const char*>(&this->_value._nextHop),sizeof(this->_value._nextHop));
      break;
    case Bgp4_Attribute_MultiExitDisc:
      g_ArtsLibInternal_Primitive.WriteUint32(os,this->_value._MED,
@@ -654,7 +654,7 @@
      break;
    case Bgp4_Attribute_Community:
      numCommunities = this->_value._community->size();
-     os.write(&numCommunities,sizeof(numCommunities));
+     os.write(reinterpret_cast<char*>(&numCommunities),sizeof(numCommunities));
      for (int commNum = 0; commNum < numCommunities; commNum++) {
        g_ArtsLibInternal_Primitive.WriteUint32(os,
                                    (*(this->_value._community))[commNum],
@@ -680,7 +680,7 @@
 //
 //----------------------------------------------------------------------------
-int ArtsBgp4Attribute::write(int fd, uint8_t version = 0) const
+int ArtsBgp4Attribute::write(int fd, uint8_t version) const
 {
   uint8_t  numCommunities;
   int      rc;
@@ -855,7 +855,7 @@
 //
 //----------------------------------------------------------------------------
-uint16_t ArtsBgp4Attribute::Length(uint8_t version = 0) const
```

```
+uint16_t ArtsBgp4Attribute::Length(uint8_t version) const
 {
   uint16_t  length = sizeof(this->_flags) + sizeof(this->_type);
   switch (this->_type) {
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4DPAttribute.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4DPAttribute.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4DPAttribute.cc 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4DPAttribute.cc 2004-01-11 19:29:22.000000000 -0800
@@ -45,6 +45,8 @@
 #include "ArtsPrimitive.hh"
 #include "ArtsBgp4DPAttribute.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsBgp4DPAttribute.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp $";

 //---------------------------------------------------------------------------
@@ -132,7 +134,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream & ArtsBgp4DPAttribute::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4DPAttribute::read(istream & is, uint8_t version)
 {
   g_ArtsLibInternal_Primitive.ReadUint16(is,this->_as,sizeof(this->_as));
   g_ArtsLibInternal_Primitive.ReadUint32(is,this->_value,
@@ -145,7 +147,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4DPAttribute::read(int fd, uint8_t version = 0)
+int ArtsBgp4DPAttribute::read(int fd, uint8_t version)
```

```
{
  int   rc;
  int   bytesRead = 0;
@@ -169,7 +171,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-ostream & ArtsBgp4DPAttribute::write(ostream & os, uint8_t version = 0) const
+ostream & ArtsBgp4DPAttribute::write(ostream & os, uint8_t version) const
 {
   g_ArtsLibInternal_Primitive.WriteUint16(os,this->_as,sizeof(this->_as));
   g_ArtsLibInternal_Primitive.WriteUint32(os,this->_value,
@@ -182,7 +184,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsBgp4DPAttribute::write(int fd, uint8_t version = 0) const
+int ArtsBgp4DPAttribute::write(int fd, uint8_t version) const
 {
   int   rc;
   int   bytesWritten = 0;
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4Prefix.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4Prefix.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4Prefix.cc       2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4Prefix.cc         2004-01-11 19:29:23.000000000 -0800
@@ -48,6 +48,8 @@


 #include "ArtsBgp4Prefix.hh"


+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsBgp4Prefix.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp
$";
```

```
//--------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4RouteEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4RouteEntry.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4RouteEntry.cc 2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4RouteEntry.cc 2004-01-11 19:29:24.000000000 -0800
@@ -122,7 +122,7 @@
 //.....
 //
 //--------------------------------------------------------------------------
-istream & ArtsBgp4RouteEntry::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4RouteEntry::read(istream & is, uint8_t version)
 {
   ArtsBgp4Attribute  bgp4Attribute;

@@ -186,7 +186,7 @@
 //.....
 //
 //--------------------------------------------------------------------------
-int ArtsBgp4RouteEntry::read(int fd, uint8_t version = 0)
+int ArtsBgp4RouteEntry::read(int fd, uint8_t version)
 {
   int               rc;
   ArtsBgp4Attribute  bgp4Attribute;
@@ -291,7 +291,7 @@
 //.....
 //
 //--------------------------------------------------------------------------
-ostream & ArtsBgp4RouteEntry::write(ostream & os, uint8_t version = 0) const
+ostream & ArtsBgp4RouteEntry::write(ostream & os, uint8_t version) const
 {
   g_ArtsLibInternal_Primitive.WriteUint32(os,this->_attrIndex,
                                           sizeof(this->_attrIndex));
@@ -309,7 +309,7 @@
```

```
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsBgp4RouteEntry::write(int fd, uint8_t version = 0) const
+int ArtsBgp4RouteEntry::write(int fd, uint8_t version) const
 {
   int   rc;
   int   bytesWritten = 0;
@@ -337,7 +337,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-uint32_t ArtsBgp4RouteEntry::Length(uint8_t version = 0) const
+uint32_t ArtsBgp4RouteEntry::Length(uint8_t version) const
 {
   uint32_t  length = sizeof(this->_attrIndex);
   vector<ArtsBgp4Attribute>::const_iterator  bgp4AttrIter;
diff -u -r arts++-1-1-a9/classes/src/ArtsBgp4RouteTableData.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsBgp4RouteTableData.cc
--- arts++-1-1-a9/classes/src/ArtsBgp4RouteTableData.cc     2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBgp4RouteTableData.cc     2004-01-11 19:29:24.000000000 -0800
@@ -114,7 +114,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream & ArtsBgp4RouteTableData::read(istream & is, uint8_t version = 0)
+istream & ArtsBgp4RouteTableData::read(istream & is, uint8_t version)
 {
   uint32_t           numRoutes;
   ArtsBgp4RouteEntry  routeEntry;
@@ -134,7 +134,7 @@
 //.....................................................................
 //
```

```
 //-----------------------------------------------------------------------------
-int ArtsBgp4RouteTableData::read(int fd, uint8_t version = 0)
+int ArtsBgp4RouteTableData::read(int fd, uint8_t version)
 {
   uint32_t            numRoutes;
   ArtsBgp4RouteEntry  routeEntry;
@@ -165,7 +165,7 @@
 //
 //-----------------------------------------------------------------------------
 ostream & ArtsBgp4RouteTableData::write(ostream & os,
-                                        uint8_t version = 0) const
+                                        uint8_t version) const
 {
   Ipv4PrefixPatricia<ArtsBgp4RouteEntry>::iterator  routeIter;

@@ -185,7 +185,7 @@
 //.....................................................................
 //
 //-----------------------------------------------------------------------------
-int ArtsBgp4RouteTableData::write(int fd, uint8_t version = 0) const
+int ArtsBgp4RouteTableData::write(int fd, uint8_t version) const
 {
   Ipv4PrefixPatricia<ArtsBgp4RouteEntry>::iterator  routeIter;
   int    rc;
@@ -216,7 +216,7 @@
 //.....................................................................
 //
 //-----------------------------------------------------------------------------
-uint32_t ArtsBgp4RouteTableData::Length(uint8_t version = 0) const
+uint32_t ArtsBgp4RouteTableData::Length(uint8_t version) const
 {
   uint32_t  length = sizeof(uint32_t);
   Ipv4PrefixPatricia<ArtsBgp4RouteEntry>::iterator  routeIter;
```

```
diff -u -r arts++-1-1-a9/classes/src/ArtsBitString.cc arts++-1-1-a9.modified.backup/classes/src/ArtsBitString.cc
--- arts++-1-1-a9/classes/src/ArtsBitString.cc          2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsBitString.cc          2004-01-11 19:29:23.000000000 -0800
@@ -48,6 +48,8 @@

 #include "ArtsBitString.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsBitString.cc,v 1.1.1.1 2002/11/16 00:46:41 rkoga Exp $";

 //----------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsFileUtil.cc arts++-1-1-a9.modified.backup/classes/src/ArtsFileUtil.cc
--- arts++-1-1-a9/classes/src/ArtsFileUtil.cc           2002-11-15 16:46:41.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsFileUtil.cc           2004-01-11 19:29:23.000000000 -0800
@@ -51,6 +51,7 @@
 #include <errno.h>
 }

+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -63,6 +64,8 @@
 #include "ArtsDebug.hh"
 #include "ArtsPackageVersion.hh"

+using namespace std;
+
 typedef map<ArtsAggregatorMapKey,time_t,less<ArtsAggregatorMapKey> >  \
 IntervalStartMap_t;

diff -u -r arts++-1-1-a9/classes/src/ArtsHeader.cc arts++-1-1-a9.modified.backup/classes/src/ArtsHeader.cc
```

```
--- arts++-1-1-a9/classes/src/ArtsHeader.cc           2003-04-28 13:02:55.000000000 -0700
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsHeader.cc             2004-01-11 19:29:22.000000000 -0800
@@ -51,6 +51,8 @@
 #include "ArtsHeader.hh"
 #include "ArtsPrimitive.hh"

+using namespace std;
+
 typedef struct {
   uint32_t     identifier;
   const char  *name;
@@ -135,23 +137,23 @@
   uint32_t  uIntDatum, idAndVersion;

   uShortDatum = htons(this->_magic);
-  os.write(&uShortDatum,sizeof(uShortDatum));
+  os.write(reinterpret_cast<char*>(&uShortDatum),sizeof(uShortDatum));

   idAndVersion = (this->_identifier << 4) | this->_version;
   uIntDatum = htonl(idAndVersion);
-  os.write(&uIntDatum,sizeof(uIntDatum));
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));

   uIntDatum = htonl(this->_flags);
-  os.write(&uIntDatum,sizeof(uIntDatum));
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));

   uShortDatum = htons(this->_numAttributes);
-  os.write(&uShortDatum,sizeof(uShortDatum));
+  os.write(reinterpret_cast<char*>(&uShortDatum),sizeof(uShortDatum));

   uIntDatum = htonl(this->_attrLength);
-  os.write(&uIntDatum,sizeof(uIntDatum));
```

```
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));


   uIntDatum = htonl(this->_dataLength);
-  os.write(&uIntDatum,sizeof(uIntDatum));
+  os.write(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));


   return(os);
 }
@@ -225,13 +227,13 @@
   uint32_t  uIntDatum;
   uint32_t  idAndVersion;

-  is.read(&uShortDatum,sizeof(uShortDatum));
+  is.read(reinterpret_cast<char*>(&uShortDatum),sizeof(uShortDatum));
   if (is.eof())
     return(is);

   this->_magic = ntohs(uShortDatum);

-  is.read(&uIntDatum,sizeof(uIntDatum));
+  is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
   if (is.eof())
     return(is);

@@ -239,22 +241,22 @@
   this->_identifier = idAndVersion >> 4;
   this->_version    = idAndVersion & 0x0f;

-  is.read(&uIntDatum,sizeof(uIntDatum));
+  is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
   if (is.eof())
     return(is);
   this->_flags = ntohl(uIntDatum);
```

```
-  is.read(&uShortDatum,sizeof(uShortDatum));
+  is.read(reinterpret_cast<char*>(&uShortDatum),sizeof(uShortDatum));
   if (is.eof())
     return(is);
   this->_numAttributes = ntohs(uShortDatum);


-  is.read(&uIntDatum,sizeof(uIntDatum));
+  is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
   if (is.eof())
     return(is);
   this->_attrLength = ntohl(uIntDatum);


-  is.read(&uIntDatum,sizeof(uIntDatum));
+  is.read(reinterpret_cast<char*>(&uIntDatum),sizeof(uIntDatum));
   if (is.eof())
     return(is);
   this->_dataLength = ntohl(uIntDatum);
Only in arts++-1-1-a9.modified.backup/classes/src: ArtsIfIndexSelectionSet.cc
diff -u -r arts++-1-1-a9/classes/src/ArtsInterfaceMatrixAggregatorMap.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsInterfaceMatrixAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsInterfaceMatrixAggregatorMap.cc       2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsInterfaceMatrixAggregatorMap.cc       2004-01-11
19:29:24.000000000 -0800
@@ -45,6 +45,7 @@
 }


 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@
```

```
 #include "ArtsInterfaceMatrixAggregatorMap.hh"


+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsInterfaceMatrixAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";


 //---------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsInterfaceMatrixData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsInterfaceMatrixData.cc
--- arts++-1-1-a9/classes/src/ArtsInterfaceMatrixData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsInterfaceMatrixData.cc 2004-01-11 19:29:21.000000000 -0800
@@ -44,6 +44,8 @@
 #include "ArtsInterfaceMatrixData.hh"
 #include "ArtsPrimitive.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsInterfaceMatrixData.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";


 //---------------------------------------------------------------------------
@@ -81,7 +83,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream & ArtsInterfaceMatrixData::read(istream& is, uint8_t version = 0)
+istream & ArtsInterfaceMatrixData::read(istream& is, uint8_t version)
 {
   uint32_t entryNum;
   ArtsInterfaceMatrixEntry interfaceEntry;
@@ -106,7 +108,7 @@
 //.....................................................................
```

```
 //
 //------------------------------------------------------------------------
-int ArtsInterfaceMatrixData::read(int fd, uint8_t version = 0)
+int ArtsInterfaceMatrixData::read(int fd, uint8_t version)
 {
   uint32_t                  entryNum;
   //  ArtsPrimitive             ioHelper;
@@ -163,7 +165,7 @@
 //
 //------------------------------------------------------------------------
 ostream & ArtsInterfaceMatrixData::write(ostream& os,
-                                         uint8_t version = 0)
+                                         uint8_t version)
 {
   uint32_t       entryNum;

@@ -185,7 +187,7 @@
 //.....................................................................
 //
 //------------------------------------------------------------------------
-int ArtsInterfaceMatrixData::write(int fd, uint8_t version = 0)
+int ArtsInterfaceMatrixData::write(int fd, uint8_t version)
 {
   uint32_t       entryNum;
   int            rc;
@@ -237,7 +239,7 @@
 //.....................................................................
 //
 //------------------------------------------------------------------------
-uint32_t ArtsInterfaceMatrixData::Length(uint8_t version = 0) const
+uint32_t ArtsInterfaceMatrixData::Length(uint8_t version) const
 {
   uint32_t  length = 0;
```

```
diff -u -r arts++-1-1-a9/classes/src/ArtsInterfaceMatrixEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsInterfaceMatrixEntry.cc
--- arts++-1-1-a9/classes/src/ArtsInterfaceMatrixEntry.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsInterfaceMatrixEntry.cc 2004-01-11 19:29:23.000000000 -0800
@@ -214,7 +214,7 @@
 //.....
 //
 //-----------------------------------------------------------------------
-uint32_t ArtsInterfaceMatrixEntry::Length(uint8_t version = 0) const
+uint32_t ArtsInterfaceMatrixEntry::Length(uint8_t version) const
 {
   uint32_t   length;
   length = (sizeof(this->_descriptor) +
@@ -231,11 +231,11 @@
 //.....
 //
 //-----------------------------------------------------------------------
-istream & ArtsInterfaceMatrixEntry::read(istream& is, uint8_t version = 0)
+istream & ArtsInterfaceMatrixEntry::read(istream& is, uint8_t version)
 {
   uint8_t       bytesize, pktsize, srcsize, dstsize;

-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
   dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -255,7 +255,7 @@
 //.....
 //
 //-----------------------------------------------------------------------
-int ArtsInterfaceMatrixEntry::read(int fd, uint8_t version = 0)
```

```
+int ArtsInterfaceMatrixEntry::read(int fd, uint8_t version)
 {
   uint8_t        bytesize, pktsize, srcsize, dstsize;
   int            rc;
@@ -307,11 +307,11 @@
 //
 //------------------------------------------------------------------------
 ostream & ArtsInterfaceMatrixEntry::write(ostream& os,
-                                          uint8_t version = 0) const
+                                          uint8_t version) const
 {
   uint8_t        bytesize, pktsize, srcsize, dstsize;

-  os.write(&this->_descriptor,sizeof(this->_descriptor));
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
   dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -331,7 +331,7 @@
 //.....................................................................
 //
 //------------------------------------------------------------------------
-int ArtsInterfaceMatrixEntry::write(int fd, uint8_t version = 0) const
+int ArtsInterfaceMatrixEntry::write(int fd, uint8_t version) const
 {
   uint8_t        bytesize, pktsize, srcsize, dstsize;
   int            rc;
diff -u -r arts++-1-1-a9/classes/src/ArtsIpPathData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsIpPathData.cc
--- arts++-1-1-a9/classes/src/ArtsIpPathData.cc      2003-06-06 17:05:09.000000000 -0700
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsIpPathData.cc        2004-01-11 19:29:23.000000000 -0800
@@ -189,7 +189,7 @@
   #endif
 }
```

```
-uint32_t ArtsIpPathData::Length(uint8_t version = 0, uint8_t flags = 0) const
+uint32_t ArtsIpPathData::Length(uint8_t version, uint8_t flags) const
 {
   uint32_t len =  sizeof(_src) + sizeof(_dst) + sizeof(_hopDistance) +
                                 sizeof(_numHops);
@@ -217,38 +217,38 @@
   return(len);
 }

-ostream& ArtsIpPathData::write(ostream& os, uint8_t version = 0,
-                                                uint8_t flags = 0)
+ostream& ArtsIpPathData::write(ostream& os, uint8_t version,
+                                                uint8_t flags)
 {
   uint32_t  timeDatum;
   uint8_t   repliedAndNumHops;

-  os.write(&this->_src,sizeof(this->_src));
-  os.write(&this->_dst,sizeof(this->_dst));
+  os.write(reinterpret_cast<char*>(&this->_src),sizeof(this->_src));
+  os.write(reinterpret_cast<char*>(&this->_dst),sizeof(this->_dst));

   if (version >= 2) {
     timeDatum = htonl(_rtt);
   } else {
     timeDatum = htonl(_rtt / 1000000);
-    os.write(&timeDatum,sizeof(timeDatum));
+    os.write(reinterpret_cast<char*>(&timeDatum),sizeof(timeDatum));
     timeDatum = htonl(_rtt % 1000000);
   }
-  os.write(&timeDatum,sizeof(timeDatum));
+  os.write(reinterpret_cast<char*>(&timeDatum),sizeof(timeDatum));
```

```
-  os.write(&this->_hopDistance,sizeof(this->_hopDistance));
+  os.write(reinterpret_cast<char*>(&this->_hopDistance),sizeof(this->_hopDistance));

   repliedAndNumHops = (this->_destinationReplied << 7) | this->_numHops;
-  os.write(&repliedAndNumHops,sizeof(repliedAndNumHops));
+  os.write(reinterpret_cast<char*>(&repliedAndNumHops),sizeof(repliedAndNumHops));

   if (version >= 1) {
     // Only version 1 has conditional reason codes.
     if (version != 1 || _destinationReplied) {
-      os.write(&_haltReason, sizeof(_haltReason));
-      os.write(&_haltReasonData, sizeof(_haltReasonData));
+      os.write(reinterpret_cast<char*>(&_haltReason), sizeof(_haltReason));
+      os.write(reinterpret_cast<char*>(&_haltReasonData), sizeof(_haltReasonData));
     }
   }
   if (version >= 2) {
-    os.write(&_replyTtl, sizeof(_replyTtl));
+    os.write(reinterpret_cast<char*>(&_replyTtl), sizeof(_replyTtl));
   }

   //  sort by hop number
@@ -263,7 +263,7 @@
   return(os);
 }

-int ArtsIpPathData::write(int fd, uint8_t version = 0, uint8_t flags = 0)
+int ArtsIpPathData::write(int fd, uint8_t version, uint8_t flags)
 {
   uint32_t  timeDatum;
   uint8_t   repliedAndNumHops;
@@ -352,40 +352,40 @@
```

```
  return(bytesWritten);
}

-istream& ArtsIpPathData::read(istream& is, uint8_t version = 0,
-                              uint8_t flags = 0)
+istream& ArtsIpPathData::read(istream& is, uint8_t version,
+                              uint8_t flags)
{
  uint32_t  timeDatum;
  uint8_t   repliedAndNumHops;
  size_t    hopNum;

-  is.read(&this->_src,sizeof(this->_src));
-  is.read(&this->_dst,sizeof(this->_dst));
+  is.read(reinterpret_cast<char*>(&this->_src),sizeof(this->_src));
+  is.read(reinterpret_cast<char*>(&this->_dst),sizeof(this->_dst));

-  is.read(&timeDatum,sizeof(timeDatum));
+  is.read(reinterpret_cast<char*>(&timeDatum),sizeof(timeDatum));
  if (version >= 2) {
    _rtt = ntohl(timeDatum);
  } else {
    _rtt = ntohl(timeDatum) * 1000000;
-   is.read(&timeDatum,sizeof(timeDatum));
+   is.read(reinterpret_cast<char*>(&timeDatum),sizeof(timeDatum));
    _rtt += ntohl(timeDatum);
  }

-  is.read(&this->_hopDistance,sizeof(this->_hopDistance));
+  is.read(reinterpret_cast<char*>(&this->_hopDistance),sizeof(this->_hopDistance));

-  is.read(&repliedAndNumHops,sizeof(repliedAndNumHops));
+  is.read(reinterpret_cast<char*>(&repliedAndNumHops),sizeof(repliedAndNumHops));
```

```
  this->_destinationReplied = repliedAndNumHops >> 7;
  this->_numHops = repliedAndNumHops & 0x7f;

  if (version >= 1) {
    // Only version 1 has conditional reason codes.
    if (version != 1 || _destinationReplied) {
-     is.read(&_haltReason, sizeof(_haltReason));
-     is.read(&_haltReasonData, sizeof(_haltReasonData));
+     is.read(reinterpret_cast<char*>(&_haltReason), sizeof(_haltReason));
+     is.read(reinterpret_cast<char*>(&_haltReasonData), sizeof(_haltReasonData));
    }
  }
  if (version >= 2) {
-   is.read(&_replyTtl, sizeof(_replyTtl));
+   is.read(reinterpret_cast<char*>(&_replyTtl), sizeof(_replyTtl));
  }

  if (this->_path.size() > 0) {
@@ -406,7 +406,7 @@
  return(is);
 }

-int ArtsIpPathData::read(int fd, uint8_t version = 0, uint8_t flags = 0)
+int ArtsIpPathData::read(int fd, uint8_t version, uint8_t flags)
 {
  uint32_t  timeDatum;
  uint8_t   repliedAndNumHops;
@@ -749,7 +749,7 @@
 //
 //--------------------------------------------------------------------------
 void ArtsIpPathData::AddHop(ipv4addr_t ipAddr, uint8_t hopNum,
-                            const struct timeval & rtt, uint8_t numTries=-1)
+                            const struct timeval & rtt, uint8_t numTries)
```

```
{
  ArtsIpPathEntry  pathEntry(ipAddr,hopNum);
  pathEntry.Rtt(rtt);
diff -u -r arts++-1-1-a9/classes/src/ArtsIpPathEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsIpPathEntry.cc
--- arts++-1-1-a9/classes/src/ArtsIpPathEntry.cc     2003-05-31 22:25:31.000000000 -0700
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsIpPathEntry.cc       2004-01-11 19:29:22.000000000 -0800
@@ -107,14 +107,14 @@
 }



-istream& ArtsIpPathEntry::read(istream& is, uint8_t version = 0,
-                                            uint8_t flags = 0)
+istream& ArtsIpPathEntry::read(istream& is, uint8_t version,
+                                            uint8_t flags)
 {
-  is.read(&this->_hopNum,sizeof(this->_hopNum));
+  is.read(reinterpret_cast<char*>(&this->_hopNum),sizeof(this->_hopNum));
   if (! is)
     return(is);


-  is.read(&this->_ipAddr,sizeof(this->_ipAddr));
+  is.read(reinterpret_cast<char*>(&this->_ipAddr),sizeof(this->_ipAddr));
   if (! is)
     return(is);


@@ -125,14 +125,14 @@
       if (! is)
             return(is);


-     is.read(&this->_numTries,sizeof(this->_numTries));
+     is.read(reinterpret_cast<char*>(&this->_numTries),sizeof(this->_numTries));
    }
  }
```

```
  return(is);
 }

-int ArtsIpPathEntry::read(int fd, uint8_t version = 0, uint8_t flags = 0)
+int ArtsIpPathEntry::read(int fd, uint8_t version, uint8_t flags)
 {
  int rc;
  int bytesRead = 0;
@@ -169,24 +169,24 @@
  return(bytesRead);
 }

-ostream& ArtsIpPathEntry::write(ostream& os, uint8_t version = 0,
-                                uint8_t flags = 0) const
+ostream& ArtsIpPathEntry::write(ostream& os, uint8_t version,
+                                uint8_t flags) const
 {
-  os.write(&this->_hopNum,sizeof(this->_hopNum));
-  os.write(&this->_ipAddr,sizeof(this->_ipAddr));
+  os.write(reinterpret_cast<char*>(const_cast<uint8_t*>(&this->_hopNum)),sizeof(this->_hopNum));
+  os.write(reinterpret_cast<char*>(const_cast<ipv4addr_t*>(&this->_ipAddr)),sizeof(this->_ipAddr));

  if (version >= 1) {
    // Version 1 always has iRTT.
    if (version == 1 || flags & k_rtt) {
      g_ArtsLibInternal_Primitive.WriteUint32(os, _rtt, sizeof(_rtt));
-     os.write(&this->_numTries,sizeof(this->_numTries));
+     os.write(reinterpret_cast<char*>(const_cast<uint8_t*>(&this->_numTries)),sizeof(this->_numTries));
    }
  }

  return(os);
```

```
}


-int ArtsIpPathEntry::write(int fd, uint8_t version = 0, uint8_t flags = 0) const
+int ArtsIpPathEntry::write(int fd, uint8_t version, uint8_t flags) const
{
  int  rc;
  int  bytesWritten = 0;
diff -u -r arts++-1-1-a9/classes/src/ArtsNetMatrixAggregatorMap.cc arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsNetMatrixAggregatorMap.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixAggregatorMap.cc 2004-01-11 19:29:22.000000000 -0800
@@ -45,6 +45,7 @@
 }


 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@


 #include "ArtsNetMatrixAggregatorMap.hh"


+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsNetMatrixAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";


 //------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsNetMatrixData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixData.cc
--- arts++-1-1-a9/classes/src/ArtsNetMatrixData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixData.cc 2004-01-11 19:29:22.000000000 -0800
@@ -83,7 +83,7 @@
```

```
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream & ArtsNetMatrixData::read(istream& is, uint8_t version = 2)
+istream & ArtsNetMatrixData::read(istream& is, uint8_t version)
 {
   uint32_t           entryNum;
   ArtsNetMatrixEntry  netEntry;
@@ -108,7 +108,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNetMatrixData::read(int fd, uint8_t version = 2)
+int ArtsNetMatrixData::read(int fd, uint8_t version)
 {
   uint32_t           entryNum;
   ArtsNetMatrixEntry  netEntry;
@@ -163,7 +163,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-ostream & ArtsNetMatrixData::write(ostream& os, uint8_t version = 2)
+ostream & ArtsNetMatrixData::write(ostream& os, uint8_t version)
 {
   uint32_t       entryNum;

@@ -185,7 +185,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNetMatrixData::write(int fd, uint8_t version = 2)
+int ArtsNetMatrixData::write(int fd, uint8_t version)
 {
```

```
  uint32_t       entryNum;
  int            rc;
@@ -237,7 +237,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-uint32_t ArtsNetMatrixData::Length(uint8_t version = 2) const
+uint32_t ArtsNetMatrixData::Length(uint8_t version) const
 {
   uint32_t  length = 0;

diff -u -r arts++-1-1-a9/classes/src/ArtsNetMatrixEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixEntry.cc
--- arts++-1-1-a9/classes/src/ArtsNetMatrixEntry.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNetMatrixEntry.cc 2004-01-11 19:29:23.000000000 -0800
@@ -186,7 +186,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-uint32_t ArtsNetMatrixEntry::Length(uint8_t version = 2) const
+uint32_t ArtsNetMatrixEntry::Length(uint8_t version) const
 {
   uint32_t   length;

@@ -203,7 +203,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream& ArtsNetMatrixEntry::read(istream& is, uint8_t version = 2)
+istream& ArtsNetMatrixEntry::read(istream& is, uint8_t version)
 {
   uint8_t       bytesize,
                 pktsize,
```

```
@@ -212,7 +212,7 @@
                dstsize,
                dstmasklen;

-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));
   this->_descriptor = ntohs(this->_descriptor);

   srcmasklen = ((this->_descriptor >> 5) & 0x1f) + 1;
@@ -237,7 +237,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNetMatrixEntry::read(int fd, uint8_t version = 2)
+int ArtsNetMatrixEntry::read(int fd, uint8_t version)
 {
   uint8_t       bytesize,
                 pktsize,
@@ -298,7 +298,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-ostream & ArtsNetMatrixEntry::write(ostream & os, uint8_t version = 2) const
+ostream & ArtsNetMatrixEntry::write(ostream & os, uint8_t version) const
 {
   uint8_t       bytesize,
                 pktsize,
@@ -309,7 +309,7 @@
   uint16_t      tmpDescriptor;

   tmpDescriptor = htons(this->_descriptor);
-  os.write(&tmpDescriptor,sizeof(tmpDescriptor));
+  os.write(reinterpret_cast<char*>(&tmpDescriptor),sizeof(tmpDescriptor));
```

```
  srcmasklen = ((this->_descriptor >> 5) & 0x1f) + 1;
  srcsize = (srcmasklen + 7) / 8;
@@ -333,7 +333,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNetMatrixEntry::write(int fd, uint8_t version = 2) const
+int ArtsNetMatrixEntry::write(int fd, uint8_t version) const
 {
   uint8_t        bytesize,
                  pktsize,
diff                -u                -r                arts++-1-1-a9/classes/src/ArtsNextHopTableAggregatorMap.cc                arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsNextHopTableAggregatorMap.cc        2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableAggregatorMap.cc          2004-01-11 19:29:22.000000000 -0800
@@ -45,6 +45,7 @@
 }

 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@

 #include "ArtsNextHopTableAggregatorMap.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsNextHopTableAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";
```

```
//---------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsNextHopTableData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableData.cc
--- arts++-1-1-a9/classes/src/ArtsNextHopTableData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableData.cc 2004-01-11 19:29:22.000000000 -0800
@@ -182,7 +182,7 @@
//.....
//
//---------------------------------------------------------------------------
-uint32_t ArtsNextHopTableData::ComputeLength(uint8_t version = 0) const
+uint32_t ArtsNextHopTableData::ComputeLength(uint8_t version) const
{
  this->_length = 0;

@@ -205,7 +205,7 @@
//.....
//
//---------------------------------------------------------------------------
-uint32_t ArtsNextHopTableData::Length(uint8_t version = 0) const
+uint32_t ArtsNextHopTableData::Length(uint8_t version) const
{
  this->ComputeLength(version);
  return(this->_length);
@@ -216,7 +216,7 @@
//.....
//
//---------------------------------------------------------------------------
-istream& ArtsNextHopTableData::read(istream& is, uint8_t version = 0)
+istream& ArtsNextHopTableData::read(istream& is, uint8_t version)
{
  uint32_t  numNextHops;
  uint32_t  NextHopNum;
@@ -239,7 +239,7 @@
```

```
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNextHopTableData::read(int fd, uint8_t version = 0)
+int ArtsNextHopTableData::read(int fd, uint8_t version)
 {
   uint32_t           numNextHops;
   uint32_t           NextHopNum;
@@ -281,7 +281,7 @@
 //
 //---------------------------------------------------------------------
 ostream& ArtsNextHopTableData::write(ostream& os,
-                                     uint8_t version = 0) const
+                                     uint8_t version) const
 {
   uint32_t           numNextHops;

@@ -306,7 +306,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsNextHopTableData::write(int fd, uint8_t version = 0) const
+int ArtsNextHopTableData::write(int fd, uint8_t version) const
 {
   uint32_t      numNextHops;
   int           rc;
diff -u -r arts++-1-1-a9/classes/src/ArtsNextHopTableEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableEntry.cc
--- arts++-1-1-a9/classes/src/ArtsNextHopTableEntry.cc  2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsNextHopTableEntry.cc  2004-01-11 19:29:24.000000000 -0800
@@ -54,6 +54,8 @@
 #include "ArtsPrimitive.hh"
 #include "ArtsNextHopTableEntry.hh"
```

```
+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsNextHopTableEntry.cc,v 1.1.1.1 2002/11/16 00:46:42
rkoga Exp $";


 //---------------------------------------------------------------------------
@@ -192,7 +194,7 @@
 //---------------------------------------------------------------------------
 //   uint32_t ArtsNextHopTableEntry::Length(uint8_t version = 0) const
 //---------------------------------------------------------------------------
-uint32_t ArtsNextHopTableEntry::Length(uint8_t version = 0) const
+uint32_t ArtsNextHopTableEntry::Length(uint8_t version) const
 {
   uint32_t  len = 0;


@@ -207,15 +209,15 @@
 //---------------------------------------------------------------------------
 // istream& ArtsNextHopTableEntry::read(istream& is, uint8_t version = 0)
 //---------------------------------------------------------------------------
-istream& ArtsNextHopTableEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsNextHopTableEntry::read(istream& is, uint8_t version)
 {
   uint8_t  fieldLen;


   //  IP address
-  is.read(&this->_ipAddr,sizeof(this->_ipAddr));
+  is.read(reinterpret_cast<char*>(&this->_ipAddr),sizeof(this->_ipAddr));


   //  descriptor
-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));
```

```
  //  pkts
  fieldLen = (this->_descriptor >> 3) + 1;
@@ -231,7 +233,7 @@
 //----------------------------------------------------------------------------
 //      int ArtsNextHopTableEntry::read(int fd, uint8_t version = 0)
 //----------------------------------------------------------------------------
-int ArtsNextHopTableEntry::read(int fd, uint8_t version = 0)
+int ArtsNextHopTableEntry::read(int fd, uint8_t version)
 {
   uint8_t  fieldLen;
   int      rc = 0;
@@ -262,15 +264,15 @@
 //                                      uint8_t version = 0) const
 //----------------------------------------------------------------------------
 ostream& ArtsNextHopTableEntry::write(ostream& os,
-                                      uint8_t version = 0) const
+                                      uint8_t version) const
 {
   uint8_t  fieldLen;

   // IP address
-  os.write(&this->_ipAddr,sizeof(this->_ipAddr));
+  os.write(reinterpret_cast<const char*>(&this->_ipAddr),sizeof(this->_ipAddr));

   // descriptor
-  os.write(&this->_descriptor,1);
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

   // pkts
   fieldLen = (this->_descriptor >> 3) + 1;
@@ -286,7 +288,7 @@
 //----------------------------------------------------------------------------
 //  int ArtsNextHopTableEntry::write(int fd, uint8_t version = 0) const
```

```
 //---------------------------------------------------------------------------
-int ArtsNextHopTableEntry::write(int fd, uint8_t version = 0) const
+int ArtsNextHopTableEntry::write(int fd, uint8_t version) const
 {
   uint8_t  fieldLen;
   int      rc = 0;
Only in arts++-1-1-a9.modified.backup/classes/src: ArtsObjectTypeSelectionSet.cc
diff -u -r arts++-1-1-a9/classes/src/ArtsPortChoice.cc arts++-1-1-a9.modified.backup/classes/src/ArtsPortChoice.cc
--- arts++-1-1-a9/classes/src/ArtsPortChoice.cc      2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortChoice.cc         2004-01-11 19:29:22.000000000 -0800
@@ -293,7 +293,7 @@
 {
   uint8_t   portLength;

-  is.read(&this->_flags,sizeof(this->_flags));
+  is.read(reinterpret_cast<char*>(&this->_flags),sizeof(this->_flags));
   if (!is)
     return(is);

@@ -364,7 +364,7 @@
 {
   uint8_t  portLength;

-  os.write(&this->_flags,sizeof(this->_flags));
+  os.write(reinterpret_cast<const char*>(&this->_flags),sizeof(this->_flags));

   portLength = 1;
   if (this->_flags & this->k_firstPortLengthMask)
diff -u -r arts++-1-1-a9/classes/src/ArtsPortChooser.lex arts++-1-1-a9.modified.backup/classes/src/ArtsPortChooser.lex
--- arts++-1-1-a9/classes/src/ArtsPortChooser.lex   2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortChooser.lex      2004-01-11 19:29:21.000000000 -0800
@@ -233,7 +233,7 @@
 //.....................................................................
```

```
 //
 //-------------------------------------------------------------------------
-uint32_t ArtsPortChooser::Length(uint8_t version = 0) const
+uint32_t ArtsPortChooser::Length(uint8_t version) const
 {
   uint32_t                          length = 0;
   vector<ArtsPortChoice>::const_iterator  choiceIter;
diff -u -r arts++-1-1-a9/classes/src/ArtsPortMatrixAggregator.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsPortMatrixAggregator.cc
--- arts++-1-1-a9/classes/src/ArtsPortMatrixAggregator.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortMatrixAggregator.cc 2004-01-11 19:29:23.000000000 -0800
@@ -327,7 +327,7 @@
 //-------------------------------------------------------------------------
 ArtsSelectedPortTable *
 ArtsPortMatrixAggregator::
-ConvertToArtsSelectedPortTable(int numTopPorts, bool byPkts = false) const
+ConvertToArtsSelectedPortTable(int numTopPorts, bool byPkts) const
 {
   ArtsPortTableEntry  portEntry;
   map<ArtsPortMatrixKeyValue,counter_t,less<ArtsPortMatrixKeyValue> >::const_iterator  portCounter;
diff -u -r arts++-1-1-a9/classes/src/ArtsPortMatrixAggregatorMap.cc arts++-1-1-
a9.modified.backup/classes/src/ArtsPortMatrixAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsPortMatrixAggregatorMap.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortMatrixAggregatorMap.cc 2004-01-11 19:29:22.000000000 -0800
@@ -45,6 +45,7 @@
 }

 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@
```

```
 #include "ArtsPortMatrixAggregatorMap.hh"


+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsPortMatrixAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";


 //--------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsPortMatrixData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsPortMatrixData.cc
--- arts++-1-1-a9/classes/src/ArtsPortMatrixData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortMatrixData.cc 2004-01-11 19:29:24.000000000 -0800
@@ -82,7 +82,7 @@
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-istream & ArtsPortMatrixData::read(istream& is, uint8_t version = 0)
+istream & ArtsPortMatrixData::read(istream& is, uint8_t version)
 {
   uint32_t             entryNum;
   ArtsPortMatrixEntry  portEntry;
@@ -107,7 +107,7 @@
 //.....................................................................
 //
 //--------------------------------------------------------------------------
-int ArtsPortMatrixData::read(int fd, uint8_t version = 0)
+int ArtsPortMatrixData::read(int fd, uint8_t version)
 {
   uint32_t             entryNum;
   ArtsPortMatrixEntry  portEntry;
@@ -162,7 +162,7 @@
 //.....................................................................
```

```
 //
 //------------------------------------------------------------------------
-ostream & ArtsPortMatrixData::write(ostream& os, uint8_t version = 0)
+ostream & ArtsPortMatrixData::write(ostream& os, uint8_t version)
 {
   uint32_t       entryNum;


@@ -184,7 +184,7 @@
 //.....................................................
 //
 //------------------------------------------------------------------------
-int ArtsPortMatrixData::write(int fd, uint8_t version = 0)
+int ArtsPortMatrixData::write(int fd, uint8_t version)
 {
   uint32_t       entryNum;
   int            rc;
@@ -236,7 +236,7 @@
 //.....................................................
 //
 //------------------------------------------------------------------------
-uint32_t ArtsPortMatrixData::Length(uint8_t version = 0) const
+uint32_t ArtsPortMatrixData::Length(uint8_t version) const
 {
   uint32_t  length = 0;


diff                -u                -r                arts++-1-1-a9/classes/src/ArtsPortMatrixEntry.cc                arts++-1-1-
a9.modified.backup/classes/src/ArtsPortMatrixEntry.cc
--- arts++-1-1-a9/classes/src/ArtsPortMatrixEntry.cc            2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortMatrixEntry.cc 2004-01-11 19:29:23.000000000 -0800
@@ -176,7 +176,7 @@
 //.....................................................
 //
 //------------------------------------------------------------------------
```

```
-uint32_t ArtsPortMatrixEntry::Length(uint8_t version = 0) const
+uint32_t ArtsPortMatrixEntry::Length(uint8_t version) const
 {
   uint32_t   length;

@@ -193,14 +193,14 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream& ArtsPortMatrixEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsPortMatrixEntry::read(istream& is, uint8_t version)
 {
   uint8_t        bytesize,
                  pktsize,
                  srcsize,
                  dstsize;

-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
   dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -220,7 +220,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsPortMatrixEntry::read(int fd, uint8_t version = 0)
+int ArtsPortMatrixEntry::read(int fd, uint8_t version)
 {
   uint8_t        bytesize,
                  pktsize,
@@ -273,14 +273,14 @@
 //.....................................................................
```

```
 //
 //------------------------------------------------------------------------
-ostream & ArtsPortMatrixEntry::write(ostream & os, uint8_t version = 0) const
+ostream & ArtsPortMatrixEntry::write(ostream & os, uint8_t version) const
 {
   uint8_t        bytesize,
                  pktsize,
                  srcsize,
                  dstsize;

-  os.write(&this->_descriptor,sizeof(this->_descriptor));
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

   srcsize = (this->_descriptor & 0x01) + 1;
   dstsize  = ((this->_descriptor >> 1) & 0x01) + 1;
@@ -300,7 +300,7 @@
 //.....................................................................
 //
 //------------------------------------------------------------------------
-int ArtsPortMatrixEntry::write(int fd, uint8_t version = 0) const
+int ArtsPortMatrixEntry::write(int fd, uint8_t version) const
 {
   uint8_t        bytesize,
                  pktsize,
diff -u -r arts++-1-1-a9/classes/src/ArtsPortTableData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsPortTableData.cc
--- arts++-1-1-a9/classes/src/ArtsPortTableData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortTableData.cc   2004-01-11 19:29:22.000000000 -0800
@@ -182,7 +182,7 @@
 //.....................................................................
 //
 //------------------------------------------------------------------------
-uint32_t ArtsPortTableData::ComputeLength(uint8_t version = 0) const
+uint32_t ArtsPortTableData::ComputeLength(uint8_t version) const
```

```
{
  this->_length = 0;

@@ -205,7 +205,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-uint32_t ArtsPortTableData::Length(uint8_t version = 0) const
+uint32_t ArtsPortTableData::Length(uint8_t version) const
 {
   this->ComputeLength(version);
   return(this->_length);
@@ -216,7 +216,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream& ArtsPortTableData::read(istream& is, uint8_t version = 0)
+istream& ArtsPortTableData::read(istream& is, uint8_t version)
 {
   uint32_t          numPorts;
   uint32_t          portNum;
@@ -239,7 +239,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsPortTableData::read(int fd, uint8_t version = 0)
+int ArtsPortTableData::read(int fd, uint8_t version)
 {
   uint32_t          numPorts;
   uint32_t          portNum;
@@ -281,7 +281,7 @@
 //
 //---------------------------------------------------------------------
```

```
 ostream& ArtsPortTableData::write(ostream& os,
-                                  uint8_t version = 0) const
+                                  uint8_t version) const
 {
   uint32_t                                numPorts;
   vector<ArtsPortTableEntry>::const_iterator  portEntry;
@@ -305,7 +305,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsPortTableData::write(int fd, uint8_t version = 0) const
+int ArtsPortTableData::write(int fd, uint8_t version) const
 {
   uint32_t      numPorts;
   int           rc;
diff                    -u                    -r                    arts++-1-1-a9/classes/src/ArtsPortTableEntry.cc                    arts++-1-1-
a9.modified.backup/classes/src/ArtsPortTableEntry.cc
--- arts++-1-1-a9/classes/src/ArtsPortTableEntry.cc               2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPortTableEntry.cc  2004-01-11 19:29:22.000000000 -0800
@@ -51,6 +51,8 @@
 #include "ArtsPrimitive.hh"
 #include "ArtsPortTableEntry.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsPortTableEntry.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga
Exp $";

 //---------------------------------------------------------------------
@@ -332,7 +334,7 @@
 //---------------------------------------------------------------------
 //      uint32_t ArtsPortTableEntry::Length(uint8_t version = 0) const
 //---------------------------------------------------------------------
```

```
-uint32_t ArtsPortTableEntry::Length(uint8_t version = 0) const
+uint32_t ArtsPortTableEntry::Length(uint8_t version) const
 {
   uint32_t  len = 0;

@@ -349,7 +351,7 @@
 //---------------------------------------------------------------------------
 //   istream& ArtsPortTableEntry::read(istream& is, uint8_t version = 0)
 //---------------------------------------------------------------------------
-istream& ArtsPortTableEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsPortTableEntry::read(istream& is, uint8_t version)
 {
   uint8_t  fieldLen;

@@ -358,7 +360,7 @@
                                     sizeof(this->_portNum));

   //  descriptor
-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   //  inPkts
   fieldLen = 1 << (this->_descriptor >> 6);
@@ -382,7 +384,7 @@
 //---------------------------------------------------------------------------
 //        int ArtsPortTableEntry::read(int fd, uint8_t version = 0)
 //---------------------------------------------------------------------------
-int ArtsPortTableEntry::read(int fd, uint8_t version = 0)
+int ArtsPortTableEntry::read(int fd, uint8_t version)
 {
   uint8_t  fieldLen;
   int      rc = 0;
@@ -421,7 +423,7 @@
```

```
//                                  uint8_t version = 0) const
//---------------------------------------------------------------------------
ostream& ArtsPortTableEntry::write(ostream& os,
-                                  uint8_t version = 0) const
+                                   uint8_t version) const
{
  uint8_t  fieldLen;

@@ -430,7 +432,7 @@
                                     sizeof(this->_portNum));

  //  descriptor
-  os.write(&this->_descriptor,1);
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

  //  inPkts
  fieldLen = 1 << (this->_descriptor >> 6);
@@ -454,7 +456,7 @@
//---------------------------------------------------------------------------
//    int ArtsPortTableEntry::write(int fd, uint8_t version = 0) const
//---------------------------------------------------------------------------
-int ArtsPortTableEntry::write(int fd, uint8_t version = 0) const
+int ArtsPortTableEntry::write(int fd, uint8_t version) const
{
  uint8_t  fieldLen;
  int      rc = 0;
diff -u -r arts++-1-1-a9/classes/src/ArtsPrimitive.cc arts++-1-1-a9.modified.backup/classes/src/ArtsPrimitive.cc
--- arts++-1-1-a9/classes/src/ArtsPrimitive.cc          2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsPrimitive.cc          2004-01-11 19:29:24.000000000 -0800
@@ -69,6 +69,8 @@

#include "ArtsPrimitive.hh"
```

```
+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsPrimitive.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp
$";


 //----------------------------------------------------------------------
@@ -150,11 +152,11 @@
   switch (len) {
     case 1:
       val8 = value;
-      os.write(&val8,sizeof(val8));
+      os.write(reinterpret_cast<char*>(&val8),sizeof(val8));
       break;
     case 2:
       val16 = htons(value);
-      os.write(&val16,sizeof(val16));
+      os.write(reinterpret_cast<char*>(&val16),sizeof(val16));
       break;
     default:
       #ifdef ARTS_DEBUG_DO_ASSERTIONS
@@ -218,11 +220,11 @@


   switch (len) {
     case 1:
-      is.read(&val8,sizeof(val8));
+      is.read(reinterpret_cast<char*>(&val8),sizeof(val8));
       value = val8;
       break;
     case 2:
-      is.read(&val16,sizeof(val16));
+      is.read(reinterpret_cast<char*>(&val16),sizeof(val16));
       value = ntohs(val16);
       break;
```

```
    default:
@@ -289,23 +291,23 @@
   switch (len) {
     case 1:
       val8 = value;
-      os.write(&val8,sizeof(val8));
+      os.write(reinterpret_cast<char*>(&val8),sizeof(val8));
       break;
     case 2:
       val16 = value;
       val16 = htons(val16);
-      os.write(&val16,sizeof(val16));
+      os.write(reinterpret_cast<char*>(&val16),sizeof(val16));
       break;
     case 3:
       val8 = (value >> 16);
-      os.write(&val8,sizeof(val8));
+      os.write(reinterpret_cast<char*>(&val8),sizeof(val8));
       val16 = htons((value & 0x0000ffff));
-      os.write(&val16,sizeof(val16));
+      os.write(reinterpret_cast<char*>(&val16),sizeof(val16));
       break;
     case 4:
       val32 = value;
       val32 = htonl(value);
-      os.write(&val32,sizeof(val32));
+      os.write(reinterpret_cast<char*>(&val32),sizeof(val32));
       break;
     default:
       #ifdef ARTS_DEBUG_DO_ASSERTIONS
@@ -384,21 +386,21 @@


   switch (len) {
```

```
    case 1:
-     is.read(&val8,sizeof(val8));
+     is.read(reinterpret_cast<char*>(&val8),sizeof(val8));
      value = val8;
      break;
    case 2:
-     is.read(&val16,sizeof(val16));
+     is.read(reinterpret_cast<char*>(&val16),sizeof(val16));
      value = ntohs(val16);
      break;
    case 3:
-     is.read(&val8,sizeof(val8));
+     is.read(reinterpret_cast<char*>(&val8),sizeof(val8));
      value = ((uint32_t)val8) << 16;
-     is.read(&val16,sizeof(val16));
+     is.read(reinterpret_cast<char*>(&val16),sizeof(val16));
      value |= ntohs(val16);
      break;
    case 4:
-     is.read(&val32,sizeof(val32));
+     is.read(reinterpret_cast<char*>(&val32),sizeof(val32));
      value = ntohl(val32);
      break;
    default:
@@ -476,29 +478,30 @@
 ostream & ArtsPrimitive::WriteUint64(ostream & os, const uint64_t & value,
                                      uint8_t len) const
 {
- uint32_t  valuePart[2];
+ const int numParts = 2;
+ uint32_t  valuePart[numParts];
   uint16_t  val16;
   uint8_t   val8;
```

```
  switch (len) {
    case 1:
      val8 = value;
-     os.write(&val8,sizeof(val8));
+      os.write(reinterpret_cast<char*>(&val8),sizeof(val8));
      break;
    case 2:
      val16 = value;
      val16 = htons(val16);
-     os.write(&val16,sizeof(val16));
+      os.write(reinterpret_cast<char*>(&val16),sizeof(val16));
      break;
    case 4:
      valuePart[0] = value;
      valuePart[0] = htonl(value);
-     os.write(&valuePart[0],sizeof(uint32_t));
+      os.write(reinterpret_cast<char*>(&valuePart[0]),sizeof(uint32_t));
      break;
    case 8:
      valuePart[0] = htonl(value >> 32);
      valuePart[1] = htonl(value & 0xffffffff);
-     os.write(valuePart,sizeof(valuePart));
+      os.write(reinterpret_cast<char*>(valuePart),numParts*sizeof(uint32_t));
      break;
    default:
      #ifdef ARTS_DEBUG_DO_ASSERTIONS
@@ -569,25 +572,26 @@
 istream & ArtsPrimitive::ReadUint64(istream & is, uint64_t & value,
                                     uint8_t len) const
 {
- uint32_t  valuePart[2];
+  const int numParts = 2;
```

```
+  uint32_t  valuePart[numParts];
   uint16_t  val16;
   uint8_t   val8;

   switch (len) {
     case 1:
-      is.read(&val8,sizeof(val8));
+       is.read(reinterpret_cast<char*>(&val8),sizeof(val8));
       value = val8;
       break;
     case 2:
-      is.read(&val16,sizeof(val16));
+       is.read(reinterpret_cast<char*>(&val16),sizeof(val16));
       value = ntohs(val16);
       break;
     case 4:
-      is.read(&valuePart[0],sizeof(uint32_t));
+       is.read(reinterpret_cast<char*>(&valuePart[0]),sizeof(uint32_t));
       value = ntohl(valuePart[0]);
       break;
     case 8:
-      is.read(valuePart,sizeof(valuePart));
+       is.read(reinterpret_cast<char*>(valuePart),numParts*sizeof(uint32_t));
       value = ((uint64_t)ntohl(valuePart[0])) << 32;
       value += ntohl(valuePart[1]);
       break;
@@ -667,24 +671,24 @@

   switch (len) {
     case 1:
-      is.read(&octet1,sizeof(octet1));
+       is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
       value = htonl((ipv4addr_t)octet1 << 24);
```

```
      break;
    case 2:
-     is.read(&octet1,sizeof(octet1));
-     is.read(&octet2,sizeof(octet2));
+      is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+      is.read(reinterpret_cast<char*>(&octet2),sizeof(octet2));
      value = htonl(((ipv4addr_t)octet1 << 24) | ((ipv4addr_t)octet2 << 16));
      break;
    case 3:
-     is.read(&octet1,sizeof(octet1));
-     is.read(&octet2,sizeof(octet2));
-     is.read(&octet3,sizeof(octet3));
+      is.read(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+      is.read(reinterpret_cast<char*>(&octet2),sizeof(octet2));
+      is.read(reinterpret_cast<char*>(&octet3),sizeof(octet3));
      value = htonl(((ipv4addr_t)octet1 << 24) |
                    ((ipv4addr_t)octet2 << 16) |
                    ((ipv4addr_t)octet3 << 8));
      break;
    case 4:
-     is.read(&ipAddr,sizeof(ipAddr));
+      is.read(reinterpret_cast<char*>(&ipAddr),sizeof(ipAddr));
      value = ipAddr;
      break;
    default:
@@ -766,26 +770,26 @@
  switch (len) {
    case 1:
      octet1 = ntohl(value) >> 24;
-     os.write(&octet1,sizeof(octet1));
+      os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
      break;
    case 2:
```

```
      netaddr = ntohl(value);
      octet1 = (ipv4addr_t)(netaddr >> 24) & 0xff;
      octet2 = (ipv4addr_t)(netaddr >> 16) & 0xff;
-     os.write(&octet1,sizeof(octet1));
-     os.write(&octet2,sizeof(octet2));
+      os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+      os.write(reinterpret_cast<char*>(&octet2),sizeof(octet2));
      break;
    case 3:
      netaddr = ntohl(value);
      octet1 = (ipv4addr_t)(netaddr >> 24) & 0xff;
      octet2 = (ipv4addr_t)(netaddr >> 16) & 0xff;
      octet3 = (ipv4addr_t)(netaddr >> 8) & 0xff;
-     os.write(&octet1,sizeof(octet1));
-     os.write(&octet2,sizeof(octet2));
-     os.write(&octet3,sizeof(octet3));
+      os.write(reinterpret_cast<char*>(&octet1),sizeof(octet1));
+      os.write(reinterpret_cast<char*>(&octet2),sizeof(octet2));
+      os.write(reinterpret_cast<char*>(&octet3),sizeof(octet3));
      break;
    case 4:
-     os.write(&value,4);
+      os.write(reinterpret_cast<const char*>(&value),sizeof(value));
      break;
    default:
      #ifdef ARTS_DEBUG_DO_ASSERTIONS
diff -u -r arts++-1-1-a9/classes/src/ArtsProtocolTableAggregatorMap.cc arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsProtocolTableAggregatorMap.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableAggregatorMap.cc 2004-01-11 19:29:23.000000000 -0800
@@ -45,6 +45,7 @@
 }
```

```
 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@

 #include "ArtsProtocolTableAggregatorMap.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsProtocolTableAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";

 //----------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsProtocolTableData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableData.cc
--- arts++-1-1-a9/classes/src/ArtsProtocolTableData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableData.cc 2004-01-11 19:29:23.000000000 -0800
@@ -182,7 +182,7 @@
 //.....................................................................
 //
 //----------------------------------------------------------------------
-uint32_t ArtsProtocolTableData::ComputeLength(uint8_t version = 0) const
+uint32_t ArtsProtocolTableData::ComputeLength(uint8_t version) const
 {
   this->_length = 0;

@@ -205,7 +205,7 @@
 //.....................................................................
 //
 //----------------------------------------------------------------------
```

```
-uint32_t ArtsProtocolTableData::Length(uint8_t version = 0) const
+uint32_t ArtsProtocolTableData::Length(uint8_t version) const
 {
   this->ComputeLength(version);
   return(this->_length);
@@ -216,7 +216,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream& ArtsProtocolTableData::read(istream& is, uint8_t version = 0)
+istream& ArtsProtocolTableData::read(istream& is, uint8_t version)
 {
   uint32_t           numProtocols;
   uint32_t           protocolNum;
@@ -239,7 +239,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsProtocolTableData::read(int fd, uint8_t version = 0)
+int ArtsProtocolTableData::read(int fd, uint8_t version)
 {
   uint32_t           numProtocols;
   uint32_t           protocolNum;
@@ -281,7 +281,7 @@
 //
 //---------------------------------------------------------------------
 ostream& ArtsProtocolTableData::write(ostream& os,
-                                      uint8_t version = 0) const
+                                       uint8_t version) const
 {
   uint32_t           numProtocols;

@@ -306,7 +306,7 @@
```

```
//.....................................................................
//
//---------------------------------------------------------------------------
-int ArtsProtocolTableData::write(int fd, uint8_t version = 0) const
+int ArtsProtocolTableData::write(int fd, uint8_t version) const
{
  uint32_t      numProtocols;
  int           rc;
diff -u -r arts++-1-1-a9/classes/src/ArtsProtocolTableEntry.cc arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableEntry.cc
--- arts++-1-1-a9/classes/src/ArtsProtocolTableEntry.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsProtocolTableEntry.cc 2004-01-11 19:29:23.000000000 -0800
@@ -51,6 +51,8 @@
 #include "ArtsPrimitive.hh"
 #include "ArtsProtocolTableEntry.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsProtocolTableEntry.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";

 //---------------------------------------------------------------------------
@@ -189,7 +191,7 @@
 //---------------------------------------------------------------------------
 //   uint32_t ArtsProtocolTableEntry::Length(uint8_t version = 0) const
 //---------------------------------------------------------------------------
-uint32_t ArtsProtocolTableEntry::Length(uint8_t version = 0) const
+uint32_t ArtsProtocolTableEntry::Length(uint8_t version) const
 {
   uint32_t  len = 0;

@@ -204,15 +206,15 @@
 //---------------------------------------------------------------------------
```

```
 // istream& ArtsProtocolTableEntry::read(istream& is, uint8_t version = 0)
 //-----------------------------------------------------------------------------
-istream& ArtsProtocolTableEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsProtocolTableEntry::read(istream& is, uint8_t version)
 {
   uint8_t  fieldLen;

   //  protocol number
-  is.read(&this->_protocolNum,sizeof(this->_protocolNum));
+  is.read(reinterpret_cast<char*>(&this->_protocolNum),sizeof(this->_protocolNum));

   //  descriptor
-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   //  pkts
   fieldLen = (this->_descriptor >> 3) + 1;
@@ -228,7 +230,7 @@
 //-----------------------------------------------------------------------------
 //      int ArtsProtocolTableEntry::read(int fd, uint8_t version = 0)
 //-----------------------------------------------------------------------------
-int ArtsProtocolTableEntry::read(int fd, uint8_t version = 0)
+int ArtsProtocolTableEntry::read(int fd, uint8_t version)
 {
   uint8_t  fieldLen;
   int      rc = 0;
@@ -259,15 +261,15 @@
 //                                      uint8_t version = 0) const
 //-----------------------------------------------------------------------------
 ostream& ArtsProtocolTableEntry::write(ostream& os,
-                                       uint8_t version = 0) const
+                                       uint8_t version) const
 {
```

```
  uint8_t  fieldLen;


  //  protocol number
-  os.write(&this->_protocolNum,sizeof(this->_protocolNum));
+  os.write(reinterpret_cast<const char*>(&this->_protocolNum),sizeof(this->_protocolNum));


  //  descriptor
-  os.write(&this->_descriptor,1);
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));


  //  pkts
  fieldLen = (this->_descriptor >> 3) + 1;
@@ -283,7 +285,7 @@
 //---------------------------------------------------------------------------
 //  int ArtsProtocolTableEntry::write(int fd, uint8_t version = 0) const
 //---------------------------------------------------------------------------
-int ArtsProtocolTableEntry::write(int fd, uint8_t version = 0) const
+int ArtsProtocolTableEntry::write(int fd, uint8_t version) const
 {
   uint8_t  fieldLen;
   int      rc = 0;
diff                -u                -r                arts++-1-1-a9/classes/src/ArtsRttTimeSeriesTableData.cc                arts++-1-1-
a9.modified.backup/classes/src/ArtsRttTimeSeriesTableData.cc
--- arts++-1-1-a9/classes/src/ArtsRttTimeSeriesTableData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsRttTimeSeriesTableData.cc   2004-01-11 19:29:23.000000000 -0800
@@ -192,7 +192,7 @@
 ostream & ArtsRttTimeSeriesTableEntry::write(ostream & os,
                                              uint32_t timeBase,
                                              uint32_t prevSecsOffset,
-                                             uint8_t version = 0) const
+                                              uint8_t version) const
 {
   uint8_t        rleFlags = 0;
```

```
  uint8_t         timestampSecsLength = 0;
@@ -217,7 +217,7 @@


  timestampUsecsLength = BytesNeededForUint32(this->_timestamp.tv_usec);
  rleFlags |= (timestampUsecsLength - 1);
-  os.write(&rleFlags,sizeof(rleFlags));
+  os.write(reinterpret_cast<char*>(&rleFlags),sizeof(rleFlags));
  if (rttLength > 0)
    g_ArtsLibInternal_Primitive.WriteUint32(os,this->_rtt,rttLength);
  if (timestampSecsLength > 0)
@@ -239,7 +239,7 @@
 int ArtsRttTimeSeriesTableEntry::write(int fd,
                                        uint32_t timeBase,
                                        uint32_t prevSecsOffset,
-                                       uint8_t version = 0) const
+                                        uint8_t version) const
 {
  uint8_t         rleFlags = 0;
  uint8_t         timestampSecsLength = 0;
@@ -302,7 +302,7 @@
 //------------------------------------------------------------------------------
 uint32_t ArtsRttTimeSeriesTableEntry::Length(uint32_t timeBase,
                                              uint32_t prevSecsOffset,
-                                             uint8_t version = 0) const
+                                              uint8_t version) const
 {
  uint32_t length = sizeof(uint8_t);  // always have rleFlags
  if (this->_rtt != k_droppedPacketRtt) {
@@ -330,13 +330,13 @@
 istream & ArtsRttTimeSeriesTableEntry::read(istream & is,
                                             uint32_t timeBase,
                                             uint32_t prevSecsOffset,
-                                            uint8_t version = 0)
```

```
+                                       uint8_t version)
 {
   uint8_t         rleFlags = 0;
   uint8_t         timestampUsecsLength = 0;
   uint32_t        timeVal;

-  is.read(&rleFlags,sizeof(rleFlags));
+  is.read(reinterpret_cast<char*>(&rleFlags),sizeof(rleFlags));

   if (rleFlags & 0x80) {
     //  it's a dropped packet; RTT length is 0
@@ -374,7 +374,7 @@
 //---------------------------------------------------------------------------
 int ArtsRttTimeSeriesTableEntry::read(int fd, uint32_t timeBase,
                                       uint32_t prevSecsOffset,
-                                      uint8_t version = 0)
+                                       uint8_t version)
 {
   uint8_t         rleFlags = 0;
   uint8_t         timestampUsecsLength = 0;
@@ -594,7 +594,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-istream & ArtsRttTimeSeriesTableData::read(istream& is, uint8_t version = 0)
+istream & ArtsRttTimeSeriesTableData::read(istream& is, uint8_t version)
 {
   uint32_t numRttEntries, rttEntryNum;
   ArtsRttTimeSeriesTableEntry  rttEntry;
@@ -623,7 +623,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
```

```
-int ArtsRttTimeSeriesTableData::read(int fd, uint8_t version = 0)
+int ArtsRttTimeSeriesTableData::read(int fd, uint8_t version)
 {
   uint32_t numRttEntries, rttEntryNum;
   ArtsRttTimeSeriesTableEntry  rttEntry;
@@ -664,7 +664,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-uint32_t ArtsRttTimeSeriesTableData::Length(uint8_t version = 0) const
+uint32_t ArtsRttTimeSeriesTableData::Length(uint8_t version) const
 {
   uint32_t    length = 0;
   uint32_t    rttEntryNum, numRttEntries;
@@ -693,7 +693,7 @@
 //
 //---------------------------------------------------------------------------
 ostream & ArtsRttTimeSeriesTableData::write(ostream & os,
-                                            uint8_t version = 0) const
+                                            uint8_t version) const
 {
   uint32_t            numRttEntries, rttEntryNum;
   uint32_t            rttUsecs;
@@ -729,7 +729,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-int ArtsRttTimeSeriesTableData::write(int fd, uint8_t version = 0) const
+int ArtsRttTimeSeriesTableData::write(int fd, uint8_t version) const
 {
   uint32_t            numRttEntries, rttEntryNum;
   uint32_t            rttUsecs;
```

```
diff -u -r arts++-1-1-a9/classes/src/ArtsSelectedPortTableData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsSelectedPortTableData.cc
--- arts++-1-1-a9/classes/src/ArtsSelectedPortTableData.cc   2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsSelectedPortTableData.cc      2004-01-11 19:29:23.000000000 -0800
@@ -188,7 +188,7 @@
 //
 //----------------------------------------------------------------------------
 uint32_t
-ArtsSelectedPortTableData::ComputeLength(uint8_t version = 0) const
+ArtsSelectedPortTableData::ComputeLength(uint8_t version) const
 {
   this->_length = 0;


@@ -214,7 +214,7 @@
 //.....................................................................
 //
 //----------------------------------------------------------------------------
-uint32_t ArtsSelectedPortTableData::Length(uint8_t version = 0) const
+uint32_t ArtsSelectedPortTableData::Length(uint8_t version) const
 {
   this->ComputeLength(version);
   return(this->_length);
@@ -227,7 +227,7 @@
 //
 //----------------------------------------------------------------------------
 istream& ArtsSelectedPortTableData::read(istream& is,
-                                         uint8_t version = 0)
+                                         uint8_t version)
 {
   uint32_t          numPorts;
   uint32_t          portNum;
@@ -251,7 +251,7 @@
 //.....................................................................
```

```
 //
 //-------------------------------------------------------------------------
-int ArtsSelectedPortTableData::read(int fd, uint8_t version = 0)
+int ArtsSelectedPortTableData::read(int fd, uint8_t version)
 {
   uint32_t            numPorts;
   uint32_t            portNum;
@@ -298,7 +298,7 @@
 //
 //-------------------------------------------------------------------------
 ostream& ArtsSelectedPortTableData::write(ostream& os,
-                                          uint8_t version = 0) const
+                                          uint8_t version) const
 {
   uint32_t                                numPorts;
   vector<ArtsPortTableEntry>::const_iterator  portEntry;
@@ -323,7 +323,7 @@
 //.....................................................................
 //
 //-------------------------------------------------------------------------
-int ArtsSelectedPortTableData::write(int fd, uint8_t version = 0) const
+int ArtsSelectedPortTableData::write(int fd, uint8_t version) const
 {
   uint32_t      numPorts;
   int           rc;
Only in arts++-1-1-a9.modified.backup/classes/src: ArtsTimeIntervalSelectionSet.cc
diff -u -r arts++-1-1-a9/classes/src/ArtsTosTableAggregatorMap.cc arts++-1-1-a9.modified.backup/classes/src/ArtsTosTableAggregatorMap.cc
--- arts++-1-1-a9/classes/src/ArtsTosTableAggregatorMap.cc2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsTosTableAggregatorMap.cc   2004-01-11 19:29:22.000000000 -0800
@@ -45,6 +45,7 @@
 }
```

```
 #include <string>
+#include <iterator>
 #ifdef HAVE_FSTREAM
   #include <fstream>
 #else
@@ -53,6 +54,8 @@

 #include "ArtsTosTableAggregatorMap.hh"

+using namespace std;
+
 static const string rcsid = "@(#) $Name: arts++-1-1-a9 $ $Id: ArtsTosTableAggregatorMap.cc,v 1.1.1.1 2002/11/16 00:46:42 rkoga Exp $";

 //---------------------------------------------------------------------------
diff -u -r arts++-1-1-a9/classes/src/ArtsTosTableData.cc arts++-1-1-a9.modified.backup/classes/src/ArtsTosTableData.cc
--- arts++-1-1-a9/classes/src/ArtsTosTableData.cc 2002-11-15 16:46:42.000000000 -0800
+++ arts++-1-1-a9.modified.backup/classes/src/ArtsTosTableData.cc    2004-01-11 19:29:22.000000000 -0800
@@ -166,7 +166,7 @@
 //---------------------------------------------------------------------------
 //   uint32_t ArtsTosTableEntry::Length(uint8_t version = 0) const
 //---------------------------------------------------------------------------
-uint32_t ArtsTosTableEntry::Length(uint8_t version = 0) const
+uint32_t ArtsTosTableEntry::Length(uint8_t version) const
 {
   uint32_t  len = 0;

@@ -179,17 +179,17 @@
 }

 //---------------------------------------------------------------------------
-// istream& ArtsTosTableEntry::read(istream& is, uint8_t version = 0)
+// istream& ArtsTosTableEntry::read(istream& is, uint8_t version)
```

```
 //---------------------------------------------------------------------------
-istream& ArtsTosTableEntry::read(istream& is, uint8_t version = 0)
+istream& ArtsTosTableEntry::read(istream& is, uint8_t version)
 {
   uint8_t  fieldLen;

   //  tos number
-  is.read(&this->_tosNum,sizeof(this->_tosNum));
+  is.read(reinterpret_cast<char*>(&this->_tosNum),sizeof(this->_tosNum));

   //  descriptor
-  is.read(&this->_descriptor,sizeof(this->_descriptor));
+  is.read(reinterpret_cast<char*>(&this->_descriptor),sizeof(this->_descriptor));

   //  pkts
   fieldLen = (this->_descriptor >> 3) + 1;
@@ -205,7 +205,7 @@
 //---------------------------------------------------------------------------
 //      int ArtsTosTableEntry::read(int fd, uint8_t version = 0)
 //---------------------------------------------------------------------------
-int ArtsTosTableEntry::read(int fd, uint8_t version = 0)
+int ArtsTosTableEntry::read(int fd, uint8_t version)
 {
   uint8_t  fieldLen;
   int      rc = 0;
@@ -236,15 +236,15 @@
 //                                      uint8_t version = 0) const
 //---------------------------------------------------------------------------
 ostream& ArtsTosTableEntry::write(ostream& os,
-                                  uint8_t version = 0) const
+                                   uint8_t version) const
 {
   uint8_t  fieldLen;
```

```
   //  tos number
-  os.write(&this->_tosNum,sizeof(this->_tosNum));
+  os.write(reinterpret_cast<const char*>(&this->_tosNum),sizeof(this->_tosNum));

   //  descriptor
-  os.write(&this->_descriptor,1);
+  os.write(reinterpret_cast<const char*>(&this->_descriptor),sizeof(this->_descriptor));

   //  pkts
   fieldLen = (this->_descriptor >> 3) + 1;
@@ -260,7 +260,7 @@
 //---------------------------------------------------------------------------
 //  int ArtsTosTableEntry::write(int fd, uint8_t version = 0) const
 //---------------------------------------------------------------------------
-int ArtsTosTableEntry::write(int fd, uint8_t version = 0) const
+int ArtsTosTableEntry::write(int fd, uint8_t version) const
 {
   uint8_t  fieldLen;
   int      rc = 0;
@@ -482,7 +482,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
-uint32_t ArtsTosTableData::ComputeLength(uint8_t version = 0) const
+uint32_t ArtsTosTableData::ComputeLength(uint8_t version) const
 {
   this->_length = 0;

@@ -505,7 +505,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------------
```

```
-uint32_t ArtsTosTableData::Length(uint8_t version = 0) const
+uint32_t ArtsTosTableData::Length(uint8_t version) const
 {
   this->ComputeLength(version);
   return(this->_length);
@@ -516,7 +516,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-istream& ArtsTosTableData::read(istream& is, uint8_t version = 0)
+istream& ArtsTosTableData::read(istream& is, uint8_t version)
 {
   uint32_t           numToss;
   uint32_t           tosNum;
@@ -539,7 +539,7 @@
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsTosTableData::read(int fd, uint8_t version = 0)
+int ArtsTosTableData::read(int fd, uint8_t version)
 {
   uint32_t           numToss;
   uint32_t           tosNum;
@@ -581,7 +581,7 @@
 //
 //---------------------------------------------------------------------
 ostream& ArtsTosTableData::write(ostream& os,
-                                 uint8_t version = 0) const
+                                 uint8_t version) const
 {
   uint32_t           numToss;

@@ -606,7 +606,7 @@
```

```
 //.....................................................................
 //
 //---------------------------------------------------------------------
-int ArtsTosTableData::write(int fd, uint8_t version = 0) const
+int ArtsTosTableData::write(int fd, uint8_t version) const
 {
   uint32_t      numToss;
   int           rc;
Only in arts++-1-1-a9.modified.backup/classes/src: Makefile
Only in arts++-1-1-a9.modified.backup: config.cache
Only in arts++-1-1-a9.modified.backup: config.log
Only in arts++-1-1-a9.modified.backup: config.status
Only in arts++-1-1-a9.modified.backup/doc: Makefile
Only in arts++-1-1-a9.modified.backup/doc: installHtml.sh
Only in arts++-1-1-a9.modified.backup/doc/libArts: Makefile
Only in arts++-1-1-a9.modified.backup/doc/libArts: installHtml.sh
Only in arts++-1-1-a9.modified.backup/include: aclocal.h
Only in arts++-1-1-a9.modified.backup: libtool
```

APPENDIX F CITATION

[1] Cowie, J., Ogielski, A.T., Premore, BJ, and Yuan Y., “Global routing instability triggered by Code Red II and Nimda worm attacks.” Extended Technical Report, Rensys Corporation, December 2001. Available from http://www.renesys.com/projects/papers/renesys_bgp_instabilities2001.pdf

[2] Monk, T., Claffy, K., “Internet Data Acquisition and Analysis: Status and Next Steps.” National Laboratory for Applied Network Research USA. Available from http://www.isoc.org/inet97/proceedings/F1/F1_3.HTM

[3] http://cnx.rice.edu/content/m10686/latest/

[4] http://www.mrtd.net/, last accessed March 2004

[5] http://www.caida.org/tools/utilities/arts/, last accessed March 2004

THIS PAGE INTENTIONALLY LEFT BLANK

BIBLIOGRAPHY

[1] Griffin, T.G., and Premore, B., “An experimental analysis of BGP convergence time,” in *Proceedings of 9th International Conference on Network Protocols*, November 2001

[2] Griffin, T.G., Gao, L., and Rexford, J., “Inherently safe backup routing with BGP,” in

Proceedings of IEEE INFOCOM, April 2001.

[3] Mahajan, R., Wetherall, D., M., and Anderson, T., “Understanding BGP misconfiguration,” in *Proceedings of SIGCOMM,* Pittsburgh, Pennsylvania, USA, August 2002.

[4] Mao, Z.M., Govindan, R., Varghese, G., and Katz, R.H., “Route Flap Damping Exacerbates Internet Routing Convergence,” in *Proceedings of SIGCOMM,* Pittsburgh, Pennsylvania, USA, August 2002.

[5] Röning, J., Laakso, M., and Takanen, A., “PROTOS: Systematic approach to eliminate software vulnerabilities,” invited presentation at Microsoft Research, Seattle, USA, May 2002.

[6] Wang, L., Zhao, X., Pei, D., Bush, R., Massey, D., Mankin, A., Wu, S.F., and Zhang, L., “Observation and analysis of BGP behavior under stress,” in *Proceedings of Internet*

Measurement Workshop, Marseille, France, November 2002.

[7] Maennel, O., and Feldmann, A., “Realistic BGP Traffic for Test Labs,” in *Proceedings of SIGCOMM,* Pittsburgh, Pennsylvania, USA, August 2002.

[8] Shaikh, A., Kalampoukas, L., Dube, R., and Varma, A., “Routing Stability in Congested Networks: Experimentation and Analysis,” in *Proceedings of SIGCOMM,* Stockholm, Sweden, August-September, 2000.

[9] Kaksonen, Rauli, “A Functional Method for Accessing Protocol Implementation Security,” Technical Report, Technical Research Centre of Finland, Oulu, Finland, September 2001.

[10] Gao, Lixin, and Rexford, Jennifer, “Stable Internet Routing without Global Coordination,” in *Proceedings of SIGMETRICS,* 2000.

[11] Liao, Y., Zhang, K., “BGP Behavior Monitoring and Analysis,” University of California, Davis, USA.

[12] Rexford, J., Wang, J., Xiao, Z., Zhang, Y., “BGP Routing Stability of Popular Destinations,” AT&T Labs-Research, Florham Park, New Jersey, USA.

[13] Feamster, N., Andersen, D., Balakrishnan, H., Kaashoek, M., “Measuring the Effects of Internet Path Faults on Reactive Routing,” in SIGMETRICS, San Diego, California, USA, June2003.

[14] http://cnx.rice.edu/content/m10673/latest/, last accessed March 2004

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Dr. Cynthia Irvine
   NPS
   Monterey, CA

4. Dr. Geoffrey Xie, Thesis Advisor
   NPS
   Monterey, CA

5. J.D. Fulp, Thesis Second Reader
   NPS
   Monterey, CA

6. Lien Nguyen
   Civilian, Naval Postgraduate School
   Monterey, CA